2

WL-TR-91-3072

LOW DENSITY REAL GAS FLOWS ABOUT HYPERSONIC VEHICLES

J. J. Hoffman
R. S. Wong
J. K. Krauss

Boeing Aerospace & Electronics
Aerospace Systems Technology CFD Group
P.O. Box 3999
Seattle, Washington 98124-2499

November 1991

Final Report for Period October 1987 to January 1991

Approved for public release, distribution is unlimited.

FLIGHT DYNAMICS DIRECTORATE
WRIGHT LABORATORY
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433-6553

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely Government-related procurement, the United States Government incurs no responsibility or any obligation whatsoever. The fact that the government may have formulated or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication, or otherwise in any manner construed, as licensing the holder, or any other person or corporation; or as conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

This report is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

This technical report has been reviewed and is approved for publication.

Denis P. Mrozinski
Project Engineer
High Speed Aero Perf Branch

Valentine Dahlem
VALENTINE DAHLEM, Chief
High Speed Aero Performance Branch
Aeromechanics Division

Robert L. Herklotz
ROBERT L. HERKLOTZ, Col, USAF
Chief
Aeromechanics Division

If your address has changed, if you wish to be removed from our mailing list, or if the addressee is no longer employed by your organization please notify WL/FIMG, WPAFB, OH 45433-6553 to help us maintain a current mailing list.

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE July 1991	3. REPORT TYPE AND DATES COVERED Final Oct 1987 - Jan 1991

4. TITLE AND SUBTITLE
Low Density Real Gas Flows About Hypersonic Vehicles

5. FUNDING NUMBERS
F33615-86-C-3006
PE 62201F
WU 2404
TA 07
WU 91

6. AUTHOR(S)
Hoffman, J. J., Wong, R. S., Krauss, J. K.

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Flight Technology CFD Group
Boeing Aerospace & Electronics
P O Box 3999
Seattle, WA 98124

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
WL/FIMG
Wright-Patterson AFB OH 45433-6553

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
WL-TR-91-3072

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release,
distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*
This document represents the final report for Air Force contract F33615-86-C-3006, "Low Density Real Gas Flows About Hypersonic Vehicles." The focus of this contract was the development of a computer code capable of predicting trends in hypersonic vehicle flowfield properties to support design trade studies for improved hypersonic vehicles. This report provides details of the Hypersonic Low Density Analysis (HYLDA) code developed for this contract. Included in this report are summaries of the HYLDA code, details of the HYLDA fully-coupled Navier-Stokes/finite rate chemistry solution algorithm, details of the thermochemistry models used by the code, and results from three Government-supplied demonstration cases. Code summaries discussed include code features, code design, and multiprocessing capability, while the algorithm discussion includes governing equations, development of the solution algorithm, and boundary conditions. The thermochemistry discussion includes details of the chemistry model transport properties model, and species thermodynamics properties model developed for the HYLDA code. The Government-supplied demonstration cases consist of a Mach 16 wedge flow, a Mach 12 biconic, and a Mach 5 catalytic hemisphere.

14. SUBJECT TERMS Hypersonic low-density flows, Navier-Stokes/finite rate chemistry algorithm, implicit algorithm with Gauss-Seidel line relaxation and flux splitting, thermochemistry models, parallel processing.

15. NUMBER OF PAGES
102

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL

PREFACE

The work reported here represents the final report for the Air Force Systems Command, Aeronautical Systems Division contract F33615-86-C-3006 "Low Density Real Gas Flows About Hypersonic Vehicles." Air Force Project Engineers were A.B. Lewis, R.D. Neumann, G.L. Burke, and D.P. Mrozinski. The work was conducted by the Aerospace Systems Technologies Computational Fluid Dynamics Group of Boeing Aerospace & Electronics, with Dr. T.C. Nark as Program Manager and Drs. J.J. Hoffman and S.F. Birch as Principal Investigators. R.S. Wong served as technical lead of the chemistry modeling phases, and J.K. Krauss and D.J. Hatch from Boeing Computer Services provided multiprocessing and programming support, respectively.

CONTENTS

FIGURES

TABLES

1.0 INTRODUCTION

This report describes the development of the Hypersonic Low Density Analysis (HYLDA) computer code. The HYLDA code is intended to support the design of complex hypersonic shapes in low density flows where real gas effects are significant. The focus of this effort was to correctly characterize the plasma state and energy distribution in the flowfield about high-velocity, high-altitude hypersonic vehicles. To do this, the HYLDA code solves the three-dimensional Navier-Stokes equations with fully-coupled finite rate air chemistry.

The development of the HYLDA code was motivated by the difficulty of current wind tunnel facilities to accurately model the shock layer around complex vehicles flying at very high altitudes and very high Mach numbers. These modeling difficulties arise from the need to accurately model nonequilibrium chemistry effects, including plasma states. The HYLDA code is capable of accounting for the effects of chemical nonequilibrium in high speed flows, including tracking the free-electron concentrations in the shock layer. AFWL is also sponsoring research directed toward upgrading wind tunnel capabilities for hypersonic flows (ref. 1).

The goal of the development effort was to provide a code capable of predicting trends in the hypersonic vehicle flowfield properties to assist in the identification of real gas effects in existing flight data, and to assist in vehicle optimization. The result of the development effort is the HYLDA code, capable of predicting heating rates, pressures, forces, and moments on complex shapes in low-density flow at hypersonic Mach numbers. The code accounts for the hypersonic flow effects of nonequilibrium air, including dissociation and ionization.

The development of the HYLDA code occurred in two phases. Phase I consisted of the development of the flow model and computational algorithm required for the calculation of nonequilibrium air chemistry flows about hypersonic vehicles. Results of the Phase I effort were reported in the Phase I Interim Technical Report (ref. 2). Phase II consisted of combining the Phase I results into the HYLDA computer code. The results of the Phase II development effort are summarized in this report.

Section 2.0 describes features of the HYLDA code, section 3.0 details the theory for the Navier-Stokes/finite rate chemistry algorithm, section 4.0 describes the thermochemistry models, and section 5.0 discusses the application of the HYLDA code to the three Government-supplied demonstration cases. Details of the use and application of the HYLDA code can be found in the Software User's Manual (ref. 3).

2.0 HYLDA CODE SUMMARY

This section provides details of the Hypersonic Low Density Analysis (HYLDA), including a summary of code features (section 2.1), a description of the code data structure (section 2.2), and a discussion of the code multiprocessing capability (section 2.3).

2.1 Code Features

A summary of the HYLDA code features is included in table 1.

Table 1. HYLDA Code Features.

• General • Versatile (multiple options) • User-friendly input	• Numerics • Arbitrarily-transformed coordinate system • Flexible boundary condition specification • Implicit Gauss-Seidel line relaxation • Inviscid flux splitting • Fully-coupled finite rate chemistry
• Physics • Full Navier-Stokes • Viscous or inviscid • Laminar or turbulent • Ideal gas, equilibrium air • Finite rate air chemistry	• Design • Modular • Flexible data structure • Dynamic memory • Vectorized • Multiprocessing • Portable

Narratives of the features in table 1 follow:

General

a. Versatile (multiple options). The HYLDA code is written to provide the user with a variety of modeling options. The options can be combined to provide analysis capability for a wide variety of flow problems.

b. User-friendly Input. The multiple options available to the user are supported by a user-friendly input generation processor, included as part of the HYLDA code. The input generation scheme provides the user with default values for each input quantity, and provides error checking and retry options to correct inputs entered in error. The input file is fully-commented, and can be edited to create new input files. Complete details of the user input process are described in detail in the Software Users Manual (ref. 3).

Physics

a. Full Navier-Stokes. The HYLDA code is based on the full Navier-Stokes equations, coupled with a one-temperature air finite rate chemistry model.

b. Viscous or inviscid. The code can be run in either a viscous or inviscid mode. The code can also be run inviscid for a number of steps before switching to a viscous calculation.

c. Laminar or turbulent. The code can be run for laminar or turbulent flows. The turbulence model used here is the Baldwin-Lomax (ref. 4) model, and is intended for use on wall-bounded turbulent flows only.

d. Ideal gas or equilibrium air. In addition to the finite rate chemistry gas model developed for this contract, the HYLDA code is also capable of calculating flows for ideal gases and for equilibrium air. The equilibrium air model used is the Tannehill model described in ref. 5.

e. Finite rate chemistry. The finite rate chemistry model included in the HYLDA code is a one-temperature model which includes a modified version of the Park-Menees (ref. 6) air chemistry model, the transport properties model of Nicolet (ref. 7), and the species thermodynamics model of McBride (ref. 8). Details of these models are included in section 4.0.

Numerics

a. Arbitrarily Transformed Coordinate System. The implicit algorithm is written in terms of generalized curvilinear coordinates, allowing body fitted meshes to be used. Body fitted meshes facilitate geometry definition (even for complex three-dimensional geometries), and also facilitate implementation of boundary conditions.

b. Flexible Boundary Condition Specification. The HYLDA code contains a full range of boundary conditions for supersonic flows, including solid wall no-slip (adiabatic or isothermal, noncatalytic or catalytic), solid wall free-slip, symmetry, inflow, outflow, and freestream conditions. These boundary conditions are treated implicitly, thereby maintaining the stability and robustness of the basic algorithm.

c. Implicit Gauss-Seidel Line Relaxation. The implicit algorithm is based on the MacCormack implicit algorithm (ref. 9). By utilizing the unfactored block tridiagonal matrix structure obtained by linearizing the implicit governing equation, the MacCormack implicit algorithm does not exhibit the time step restrictions found in approximate factorization techniques. The block tridiagonal is solved using Gauss-Seidel (G-S) line relaxation, solving a tridiagonal matrix for lines along a prescribed direction. G-S sweeps update the solution in the remaining two directions. Convergence for each time step is obtained in approximately two to three iterations (sweeps) due in part to the use of inviscid flux vector splitting which enhances the stability of the solution procedure.

d. Inviscid Flux Splitting. The HYLDA algorithm incorporates inviscid flux splitting to more accurately model flow physics than empirical methods using smoothing factors. The code uses a combination of the MacCormack (ref. 9) and Steger/Warming (ref. 10) schemes to maintain stability across strong shocks.

e. Fully-coupled finite rate chemistry. The HYLDA code solves the finite rate air chemistry flowfield about hypersonic bodies using the Navier-Stokes equations coupled to a one-temperature finite rate air chemistry model. Fully-coupled equation sets provide the most accurate modeling of the hypersonic flowfield, but are limited by the size of the matrix equations to be solved as the number of gas species increases. The HYLDA air chemistry model contains 9 species, requiring the solution of block 13 by 13 matrices for three-dimensional (3D) flows.

Design

a. Modular. The HYLDA code is built using a modular top-down structure to facilitate the incorporation of new technology as it becomes available.

b. Flexible Data Structure. The flexible data structure uses only the storage needed for a particular application. Flexibility is achieved using a pointer mapping system which allows code enhancements with a minimum of code changes.

c. Dynamic Memory. The HYLDA code supports the request field length capability on Cray and Cyber computers, and allocates only the field length needed for a particular application.

d. Vectorized. Subroutines in HYLDA have been vectorized to increase code efficiency.

e. Multiprocessing. The HYLDA code is multiprocessor compatible.

f. Portable. The HYLDA code is written in standard Fortran to facilitate porting the code to the many minisuper- and super-computers available. The Software Users Manual (ref. 3) details features of the code that are machine-specific so that the user can elect to operate selected machine-specific features if desired.

Additional details regarding general code features can be found in the Software Users Manual (ref. 3). Additional details of the physical models used in the code are included in section 4.0. Additional details regarding the numerics of the HYLDA implicit algorithm are included in section 3.0, and details of the code design are included below.

2.2 Design Features

This section provides details of the code design features summarized in table 1 and described above. A complete description of the code design can also be found in ref. 11.

Traditionally, engineering codes have been built around a solution algorithm. When the algorithm became obsolete, the code was discarded and a new code was built around a new algorithm. With the advent of supercomputers with parallel processing and vectorization and micro- and macro-processing, some codes have been built around a particular computer architecture to optimize vectorization, or to take advantage of the parallel architecture. Again, when the architecture becomes obsolete, the code may have to be discarded. Solution algorithms and computer architectures often change very rapidly, while code development has traditionally been a very slow process.

The current effort was directed at improving the ability of the code development process to rapidly and efficiently incorporate new advances in algorithms and computer architectures without requiring major code modifications. The ability to rapidly incorporate solution algorithm and computer architecture advances is accomplished through a modular code structure with a compact, flexible data structure. The ability to rapidly exploit advances in computer architecture is accomplished by requiring new coding to conform to strict standards of Fortran, without extensions, to ensure that the base version is readily portable to any computer architecture. As new architectures become available, a version of the HYLDA code can be adapted to the new architecture, while the basic version remains portable. However, high levels of vectorization can still be achieved even while adhering to the basic standards of Fortran.

Code Structure. One of the goals of the HYLDA code development effort was to produce a code that can rapidly and efficiently incorporate changes in solution algorithms while at the same time being able to exploit advances in computer architectures. The code structure chosen to satisfy the above requirements is based on a modular code design with a compact, flexible data structure. Details of the conceptual design of the code structure employed here are given by

Hopcroft (ref. 12).

A simplified schematic of the modular code design is shown in fig. 1. The controller acts as the executive driver for the code, and contains the coding to create and read input files, as well as to allocate and deallocate zones. Through the input file, the controller controls the selection of the solution aigorithm and the chemistry model.

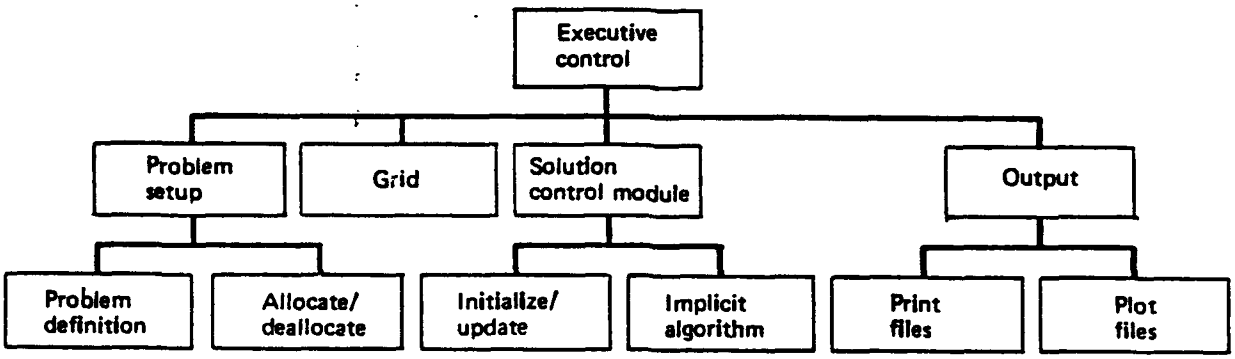


Figure 1. Top-Down Code Structure

The algorithm section contains one or more algorithms, each with a consistent interface to the controller, minimizing the changes required to install new algorithm modules. For the current development process, the algorithm section contains both the MacCormack explicit and implicit algorithms (ref. 9) extended to include finite rate chemistry effects.

The chemistry section contains routines to provide the flowfield thermodynamics. Again, all routines interface the controller, thereby making it simple to interchange chemistry routines. At present, the chemistry section contains three modules: (1) ideal gas, (2) equilibrium air (ref. 5) and (3) one-temperature finite rate chemistry.

The finite rate chemistry package consists of reaction paths, species thermodynamic properties, and species and mixture transport properties. The data requirements for these three modules have been generalized to facilitate the rapid interchange of models. Flexibility is most important in the area of finite rate chemistry, as the reaction models can vary from flow to flow, while the species thermodynamic and transport properties methods may only be suitable for certain temperature ranges and must therefore be capable of being replaced or supplemented. Details of the models included in the HYLDA code are found in section 4.0.

Flexible Data Structure. The modular code structure described above is complemented by a compact, flexible data structure which permits code modifications to be incorporated without changing the basic data structure of the code. A three-dimensional program with a wide range of options, such as the HYLDA code, requires dynamic storage for effective use of computer memory. It is impractical to statically dimension arrays with large numbers, or to expect the user to change the dimensions for each problem. Since the Fortran77 standard does not support dynamic arrays, it was necessary to create the equivalent of dynamic arrays through the use of pointers. Real, integer, and character data are maintained in separate arrays. These arrays contain all the data stored by the code.

Each of the data arrays is divided into sections. The length of each section depends on the size of the problem. The code automatically determines the length of each section, and establishes pointers to these sections. The code contains five data arrays, denoted IQ, RQR, RQ, CQ, and CQR.

The IQ array contains all of the integer data required by the code, including the pointers to the sections of all the data arrays. The IQ array has seven sections:

- Basic data,
- Boundary condition data,
- Turbulence model data,
- Reaction path data,
- Species data,
- Transport property data,
- Convergence tracking data.

In addition to the section pointers, the "basic data" contains information pertaining to the computational domain and the solution options chosen during the input creation process.

The RQR array contains real reference variables. The RQR array contains the same sections as the IQ array. For the RQR array, the "basic data" consists of the reference velocity, pressure, temperature, and time step data.

The RQ array stores the flowfield data at each cell. In each cell, the RQ array is divided into five sections:

- Nonconservative variables,
- Metrics,
- Thermodynamics,

- Solution vector,
- Additional variables.

Nonconservative variables include grid coordinates and time step for each cell, in addition to the velocity components and internal energy. Mixture thermodynamics includes pressure, temperature, speed of sound, viscosity, thermal conductivity, and specific heat ratio. Additional variable array space is included for variables not used in all models, or for new variables being tested for inclusion in the permanent arrays.

The CQ array contains character*16 data, while the CQR array contains character*72 data. The format of these arrays is the same as for the IQ array.

The arrays described above constitute the entire data structure for the code. The use of these arrays facilitates the top-down code structure by minimizing the number of arrays that must be passed in subroutine calls. Additionally, this array structure eliminates the confusion caused by excessive variable names, resulting in a compact, consistent coding scheme.

2.3 Multiprocessing Capability

The HYLDA code can be run in either a single-processor or a multi-processor mode. The single-processor mode has been written in standard Fortran for ease of portability between machines. The single-processor code has available a number of machine-specific statements which can be activated at the discretion of the user. Details of the machine-specific coding are included in the Software Users Manual (ref. 3).

The multi-processor capability of HYLDA is obtained via portable macros which are expanded into machine-specific coding statements. To maintain code modularity and portability, code parallelization efforts were restricted to schemes that did not alter the algorithm itself, but only altered the ordering of the Gauss-Seidel iterative solution process. In this manner, future changes to the algorithm will not require reworking of the multiprocessing scheme.

The baseline implicit algorithm for the code contains Gauss-Seidel solution sweeps across the computational domain. For example, the j-direction implicit line is swept in the i and k directions via standard Fortran DO loops. In their natural ordering, these sweeps are not parallelizable (because of the use of the most recently updated values at i-1 and k-1 in the G-S sweeps), but they do offer a significant amount of parallelism when reorganized into a "red-black" iteration scheme. In the red-black iteration scheme, each diagonal in the i-k plane represents independent computations that can be parallelized, although successive diagonals retain dependence on previously updated values at i-1 and k-1.

The method selected to parallelize the algorithm creates parallel processes that remain viable between iteration sweeps, with the required parameters for each new sweep obtained from shared memory. The disadvantage to this approach is that some processes will be idled while waiting for syncronization points. For large computational problems, though, the amount of idle time is not significant. Additional details regarding the parallelization of the HYLDA code can be found in ref. 13.

Code parallelization occurred in two steps: (1) code restructuring and (2) installing parallel constructs. Code restructuring required reordering the HYLDA loop structure in the implicit solver to take advantage of parallelism, and removing the loop structure to a separate subroutine. Installation of the parallel constructs included placing shared variables in shared memory, identifying (and not sharing) local variables, spawning slave processes and providing a means for scheduling work between the available CPUs.

Code Restructuring. In the original HYLDA code, implicit tridiagonal solves were performed along a j-direction line, sweeping along the k-direction for a given station in the i-direction. This ordering does not allow for any parallelism since each successive tridiagonal solve is dependent on the preceding column. To take advantage of the parallelism in the algorithm, these computations were reordered into diagonal wave fronts. Each computation along a wave front is then independent of all others in the same diagonal.

Separating the implicit solver loop into a separate subroutine was necessary because the parallel model being used here is based on a subroutine level of parallelism. That is, separate copies of a subroutine are established on "slave" processors. All variables needed as input or output must be included in either the calling sequence or in common. Common has many advantages for the parallel environment in the sharing of memory, but for this application the use of common was severely restricted.

Two options were available for the operation of the new subroutine - it could either perform one tridiagonal solve and return to ask for the next one, or it could solve the an entire wavefront iterating down the diagonal itself. Tests determined that the second option was more efficient, so the new subroutine was written that would solve an entire wavefront.

Installing Parallel Constructs. The work to be parceled out consisted of individual tridiagonal solves along each wavefront. Slaves were to be created once, and then when a wavefront is ready, they must access an available index along the wave front, until all indices have been exhausted. Once a wavefront has been completed, the slaves must wait until the next wavefront is ready, with the master process initiating the next wave front. This process is performed by a self-scheduling loop on the column subscript combined with a barrier.

In the context of multitasking, there are four classes of variables:

- Input constants,
- Input variables that may change from iteration to iteration,
- Shared input/output variables,
- Variables which are local to the current iteration.

Input constants may be passed through the calling sequence. Input variables which may change during program execution, as well as shared variables must be placed in shared common. Local variables do not need to be referenced outside the subroutine, and need not be passed in the calling sequence or in shared common.

Two different sets of tools were investigated for installation of parallel constructs: the Cray multitasking library, and monitor macros. The Cray multitasking library is a set of subprograms implemented for parallel computing on the Cray. They have also been implemented on the Alliant. Monitor macros consist of a set of macros which expand out to code that implements parallel control at a somewhat higher level than the Cray multitasking library. Instead of the user issuing scheduling commands, the user provides a simple scheduling rule as well as a location to perform the scheduling, and the monitor actually performs the scheduling. The monitor distributes tasks according to the user-specified rule.

Monitors are built from four monitor primitives. A user can use prefabricated monitors from the monitor tool set, or build new monitors using the primitives. Porting the monitors to other architectures then requires implementing the four monitor primitives on the new architecture. Since the monitors are macros, they require expansion. The current implementation of the monitors is based on the macro preprocessor, m4. The m4 macro preprocessor is a standard Unix utility.

The simplest monitor is GETSUB, which implements self-scheduling for a single DO loop. Since the HYLDA code required more than a single DO loop, it was necessary to use the ASKFOR monitor, which offers substantially more flexibility. The ASKFOR monitor allows the user to specify the rule for assigning tasks. Further, it allows for control of a set of tasks, each of which may be decomposed into subtasks.

Finally, the constructs to make the code run in parallel were installed. In the main program, all shared variables and the monitor were initialized. In the implicit solver, it was necessary to create the slaves, initialize the wavefronts, and invoke the parallel processing subroutine. In the parallel processing subroutine, it was necessary only to execute the ASKFOR monitor. The

ASKFOR monitor then schedules all work along the current wavefront.

Results. The parallelized version of the HYLDA code was created following the process described above, and a simple test case was calculated to demonstrate the parallelization. The monitor macros are truly portable, with the same code running on Alliant, Sequent, and Cray machines. Results showed that at least 60% of the code was parallelized.

The following results were achieved running the HYLDA code using a wedge test case with a 15x5x15 grid. Larger grids were not attempted because of limited storage on the available computers, and because of time constraints. The 15x5x15 grid provided the parallel algorithm wavefronts of size 15x15, while solving tridiagonals of length 5. In the parallel format, this structure produced 29 wave fronts per Gauss-Seidel sweep, with wavefronts ranging from length 1 to 15 and then back to 1. With longer wavefronts, such as those expected with more realistic problems, the parallel speedups should be greater than those demonstrated here for this small test problem.

The data were collected from two sources. The data for overall performance were collected from the real-time output from the Unix "time" command, and are used to reflect the CPU time for the entire program. The parallel loop performance data were collected from timing statements internally inserted before and after each update to the data set. These numbers reflect the wall clock time required to execute the portion of the code which ran in parallel. The ratios in the tables below were computed using the following formulae:

$$\text{Speedup} = \frac{\text{time for 1 CPU}}{\text{time for } n \text{ CPUs}} \ , \qquad (2\text{-}1)$$

$$\text{Efficiency} = \frac{\text{Speedup}}{n} \times 100\% \ . \qquad (2\text{-}2)$$

Table 2. Alliant Timing Results

Number of	Overall performance		Parallel loop performance	
CPUs	Speedup	Efficiency (%)	Speedup	Efficiency (%)
1	1.00	100	1.00	100
2	1.53	77	0.88	44
3	1.92	64	2.56	85
4	2.16	54	3.14	79
5	2.35	47	3.76	73
6	2.44	41	4.04	67
7	2.52	36	4.27	61
8	2.56	32	4.72	59

Table 3. Sequent Timing Results

Number of	Overall performance		Parallel loop performance	
CPUs	Speedup	Efficiency (%)	Speedup	Efficiency (%)
01	1.00	100	1.00	100
2	1.60	80	1.86	93
3	1.99	66	2.59	86
4	2.26	57	3.22	80
5	2.47	49	3.80	76
6	2.61	44	4.15	69

Note that in table 2 above, the two-CPU speedup for the parallel loop is less than 1, indicating a loss in performance, but that the overall performance does not differ significantly. The overall performance reflects the CPU time spent by the main processor, and table 2 shows that the addition of the second CPU did reduce the amount of time that the main processor spent on the problem. The poor 2 CPU parallel loop performance on the Alliant was caused by a problem with the Alliant operating system which permitted only one of the two CPUs to be active at a time. Further investigation of this anomaly was beyond the scope of the present work. Note, though, that the parallel performance returned to acceptable levels for the three-CPU case.

3.0 COMPUTATIONAL ALGORITHM

This section describes the computational algorithm used by the HYLDA code. The algorithm solves the full Navier-Stokes equations using the MacCormack implicit solution algorithm (ref. 9), extended under this contract to include a fully-coupled one-temperature finite rate chemistry model. Solution of the full Navier-Stokes equations, rather than reduced forms of the Navier-Stokes equations, is required for this effort since the primary interest is in the physics of the viscous shock layer surrounding hypersonic vehicles.

The decision to model the finite rate chemistry effects using a one-temperature model was based on the fact that for the flows to be considered here (up to Mach 25), one-temperature models have been shown to be sufficient to model the plasma state around hypersonic vehicles, as discussed in section 4.0. The increased accuracy afforded by multi-temperature models was not significant enough to offset the increased computational costs associated with the additional solution equations, therefore a one-temperature model was selected for the HYLDA code.

The remainder of this section details the HYLDA computational algorithm. Section 3.1 presents the governing equations, section 3.2 discusses the computational algorithm, and section 3.3 details the boundary conditions included in the code.

3.1 Governing Equations

Using a one-temperature finite rate chemistry model allows the algorithm development to follow a general procedure using the reacting gas equations, since the ideal gas and equilibrium air assumptions are special cases of the one-temperature reacting gas case. The following algorithm development is presented in terms of the finite rate chemistry algorithm. Ideal gas and equilibrium air reductions will be discussed where necessary.

The three-dimensional (3D) fully-coupled Navier-Stokes/finite rate chemistry equations consist of conservation equations for species mass, x-, y-, and z-momentum, and total energy. These equations can be written in strong conservation form in terms of physical (x,y,z) coordinates as follows:

$$\frac{\partial U'}{\partial t} + \frac{\partial F'}{\partial x} + \frac{\partial G'}{\partial y} + \frac{\partial H'}{\partial z} = \dot{W}' \quad , \qquad (3\text{-}1)$$

where $U' = \left[\rho_1, \ldots, \rho_s, \ldots, \rho_{N_s}, \rho u, \rho v, \rho w, e\right]^T$ represents the vector of conserved solution variables, $\dot{W}' = \left[\dot{\omega}_1, \ldots, \dot{\omega}_s, \ldots, \dot{\omega}_{N_s}, 0,0,0,0\right]^T$ represents the vector of chemistry source terms, and **F′**, **G′**, and **H′** represent the x-, y-, and z-direction flux vectors. Here, ρ_s denotes the density of species s, (u,v,w) represent the cartesian components of velocity, ρ represents the total density (obtained by summing the species densities), e represents the total energy per unit volume, and $\dot{\omega}_s$ represents the species s production/depletion rate, obtained from the law of mass action. In the above equations, primes denote physical (nontransformed) vectors, and T represents a vector transpose, and N_s denotes the number of species present.

The flux vectors **F′**, **G′**, and **H′** are written as:

$$\mathbf{F}'=\begin{bmatrix} \rho_1(u+\hat{u}_1) \\ \cdot \\ \cdot \\ \cdot \\ \rho_s(u+\hat{u}_s) \\ \cdot \\ \cdot \\ \cdot \\ \rho_{N_s}(u+\hat{u}_{N_s}) \\ \rho u^2+p+\tau_{xx} \\ \rho u+\tau_{xy} \\ \rho uw+\tau_{xz} \\ u(e+p)+u\tau_{xx}+v\tau_{xy}+w\tau_{xz}-\lambda\frac{\partial T}{\partial x}+\sum_{s=1}^{N_s}\rho_s\hat{u}_s h_s \end{bmatrix} \tag{3-2a}$$

$$\mathbf{G}'=\begin{bmatrix} \rho_1(v+\hat{v}_1) \\ \cdot \\ \cdot \\ \cdot \\ \rho_s(v+\hat{v}_s) \\ \cdot \\ \cdot \\ \cdot \\ \rho_{N_s}(v+\hat{v}_{N_s}) \\ \rho vu+\tau_{yx} \\ \rho v^2+p+\tau_{yx} \\ \rho vw+\tau_{yz} \\ v(e+p)+u\tau_{yx}+v\tau_{yy}+w\tau_{xz}-\lambda\frac{\partial T}{\partial y}+\sum_{s=1}^{N_s}\rho_s\hat{v}_s h_s \end{bmatrix} \tag{3-2b}$$

$$H' = \begin{bmatrix} \rho_1(w+\hat{w}_1) \\ \cdot \\ \cdot \\ \cdot \\ \rho_s(w+\hat{w}_s) \\ \cdot \\ \cdot \\ \cdot \\ \rho_{N_s}(w+\hat{w}_{N_s}) \\ \rho wu+\tau_{zx} \\ \rho vw+\tau_{zy} \\ \rho w^2+p+\tau_{zz} \\ w(e+p)+u\tau_{zx}+v\tau_{zy}+w\tau_{zz}-\lambda\frac{\partial T}{\partial z}+\sum_{s=1}^{N_s}\rho_s\hat{w}_s h_s \end{bmatrix} \qquad (3\text{-}2c)$$

where ρ_s is the species s mass density, $\rho=\sum_{s=1}^{N_s}\rho_s$ = total density, N_s = total number of species, u,v,w = cartesian components of velocity, $\hat{u}_s, \hat{v}_s, \hat{w}_s,$ = cartesian components of species diffusion velocity, p = pressure, T = temperature, e = total energy, λ = thermal conductivity, h_s = species enthalpy, and τ_{ij} is the shear stress tensor.

For the ideal gas or equilibrium air assumptions, the N_s species continuity equations reduce to a single equation for the total density, and the diffusion velocity terms $(\hat{u}_s, \hat{v}_s, \hat{w}_s)$ are zero.

The shear stress matrix is given in tensor notation as:

$$\tau_{ij} = -\mu\left[\frac{\partial q_i}{\partial x_j} + \frac{\partial q_j}{\partial x_i}\right] - \mu_B \frac{\partial q_i}{\partial x_i}\delta_{ij}, \qquad (3\text{-}3)$$

where μ is viscosity, q is the vector of cartesian velocity components, μ_B is the mixture bulk viscosity (taken here as $-\frac{2}{3}\mu$), and δ is the Kronekar delta.

The species diffusion velocities are determined from (ref. 14):

$$\rho_s\hat{q}_s = -\rho D_{s_{mix}} \frac{\partial X_s}{\partial x_j}, \qquad (3\text{-}4)$$

where $\hat{q}_s$ represents any of the three diffusion velocities, $D_{s_{mix}}$ represents the mixture diffusion coefficient of species s, X_s denotes the species molar concentration, and x_j represents the coordinate direction corresponding to the diffusion velocity being calculated. As noted above, for ideal gases and equilibrium air, the diffusion velocities are zero.

For finite rate air chemistry calculations, the viscosity (μ) and thermal conductivity (λ) are determined using transport property curve fits. Details of these models are included in section 4.0. For non-reacting gases, viscosity is obtained from Sutherland's law, and thermal conductivity is related to the viscosity using the Prandtl number.

Pressure is related to the density and temperature using Dalton's Law:

$$p = \sum_{s=1}^{N_s} \frac{\rho_s}{M_s} \hat{R} T, \tag{3-5}$$

where N_s is the total number of species (for an ideal gas $N_s = 1$), M_s is the molecular weight of species s, and $\hat{R}$ is the universal gas constant.

Temperature is obtained from the internal energy per unit mass (e_i). The internal energy per unit mass for each species may be written in terms of its translational ($e_{i_{trans}}$), rotational ($e_{i_{rot}}$), vibrational ($e_{i_{vib}}$), electronic ($e_{i_{elec}}$), and heat of formation (e_i^o) contributions:

$$e_{i_{total}} = e_{i_{trans}} + e_{i_{rot}} + e_{i_{vib}} + e_{i_{elec}} + e_i^o . \tag{3-6}$$

Solving equation (3-6) for the characteristic temperatures associated with each form of internal energy described above would require $2{*}N_s + 1$ equations (assuming the translational and rotational temperatures to be equal). Wong (ref. 15) notes that "...up to 15000K, multiple temperature models do not appear to have demonstrated sufficient improvements over one-temperature models to justify the additional complexity and computational cost" of using such methods. For the present code development effort, a one-temperature model has been assumed. That is, all internal energy modes are assumed to be a function of a single temperature. Under the one-temperature assumption, equation (3-6) can be rewritten in the following form:

$$e_{i_s} = f_s(T)T + h_s^o, \tag{3-7}$$

where $f_s(T)$ for each species is developed from curve-fit formulations for c_p using a single temperature, and h_s^o is the heat of formation of species s. For ideal gases, the heat of formation is zero, and the $f_s(T)$ terms becomes the specific heat at constant volume (c_v).

The total internal energy is obtained from:

$$e_i = \sum_{s=1}^{N_s} \frac{\rho_s}{\rho} e_{i_s} = \sum_{s=1}^{N_s} \frac{\rho_s}{\rho} f_s(T)T + \sum_{s=1}^{N_s} \frac{\rho_s}{\rho} h_s^o = f(T)T + h_f^o. \qquad (3\text{-}8)$$

Temperature is obtained from equation (3-8) directly for ideal gases (with $f(T) = c_v$), and iteratively for a reacting gas. The equilibrium air model calculates temperature as a curve-fit function of internal energy and density.

The governing equations in the HYLDA code are those described in this section in terms of a one-temperature reacting gas. Equations appropriate for ideal gases and equilibrium air are subsets of this general equation set. When applying the HYLDA code to an ideal gas case or to a case using the equilibrium air model, the appropriate governing equations are obtained through the use of option flags in the input file.

Transformed Governing Equations. To provide accurate resolution of flow gradients for complex flowfields, computational grids must be clustered near large flow gradients such as those found in shocks, shear layers, or in viscous regions near bodies. Grid clustering results in nonuniform grid spacings that are difficult to incorporate into a solution algorithm based on the physical coordinate system. The standard approach is to perform a generalized coordinate transformation on the governing equations to transform the governing equations from the physical (x,y,z) system into a body-fitted (ξ,η,ζ) system. The body-fitted coordinate system has uniform spacings in the computational domain, facilitating the development of the solution algorithm without need to account for nonuniform grid spacings.

The transformed governing equations are still written in strong conservation form:

$$\frac{\partial U}{\partial t} + \frac{\partial F}{\partial \xi} + \frac{\partial G}{\partial \eta} + \frac{\partial H}{\partial \zeta} = \dot{W} \quad , \qquad (3\text{-}9)$$

where the transformed (non-primed) fluxes are defined in terms of the physical (primed) fluxes as:

$$\mathbf{U} = J^{-1}\mathbf{U}' \quad ,$$

$$\mathbf{F} = J^{-1}\left\{\mathbf{F}'\frac{\partial\xi}{\partial x} + \mathbf{G}'\frac{\partial\xi}{\partial y} + \mathbf{H}'\frac{\partial\xi}{\partial z}\right\} \quad ,$$

$$\mathbf{G} = J^{-1}\left\{\mathbf{F}'\frac{\partial\eta}{\partial x} + \mathbf{G}'\frac{\partial\eta}{\partial y} + \mathbf{H}'\frac{\partial\eta}{\partial z}\right\} \quad , \qquad (3\text{-}10)$$

$$\mathbf{H} = J^{-1}\left\{\mathbf{F}'\frac{\partial\zeta}{\partial x} + \mathbf{G}'\frac{\partial\zeta}{\partial y} + \mathbf{H}'\frac{\partial\zeta}{\partial z}\right\} \quad ,$$

$$\dot{\mathbf{W}} = J^{-1}\dot{\mathbf{W}}' \quad ,$$

where J is the Jacobian of the transformation, with primed vectors as defined in eq. (3-1) and (3-2).

The implicit algorithm used by the HYLDA code is based on the governing equations as described above. The next two sections describe the development of the solution algorithm.

3.2 Algorithm Development

The algorithm used to solve the governing equations (3-9, 3-10) is an extension of the MacCormack block tridiagonal implicit algorithm with Gauss-Seidel line relaxation and flux splitting (ref. 9). This contract has extended the algorithm to include a fully-coupled one-temperature finite rate reacting air chemistry model containing an arbitrary number of chemical species.

Solving the equation set in fully-coupled implicit form represents the most physically correct representation of the flow physics, and should alleviate the stiffness problem often associated with alternate methods of solving the governing equations in the presence of finite rate chemistry (ref. 16). The fully-coupled approach will be computationally intensive if a large number of chemical species are present. For most applications, the number of species considered is small (less than 15 species). Considering the computing power of current supercomputers, the increased computing time required to solve the fully-coupled system should be offset by the increased convergence rates obtained by the fully-coupled implicit treatment of the equation set.

The use of a fully-coupled one-temperature reacting air chemistry model in HYLDA allows the solution algorithm to be written in general terms applicable not only to reacting gas flows, but also to ideal gas and equilibrium flows without modifying the basic code. Selection of the exact form of the algorithm (ideal gas, equilibrium air, or reacting air) is accomplished by the user through the input file.

The solution algorithm will be developed here using the full equation set, with reductions for non-reacting gas cases noted where appropriate.

The implicit algorithm development begins by recasting the governing equations (3-9), (3-10) into delta form. The first step is to split the momentum fluxes into their constituent inviscid ($\mathbf{F}_i,\mathbf{G}_i,\mathbf{H}_i$) and viscous ($\mathbf{F}_v,\mathbf{G}_v,\mathbf{H}_v$) parts. Substituting the split fluxes into (3-9) yields:

$$\left[\frac{\partial \mathbf{U}}{\partial t}+\frac{\partial \mathbf{F}_i}{\partial \xi}+\frac{\partial \mathbf{G}_i}{\partial \eta}+\frac{\partial \mathbf{H}_i}{\partial \zeta}\right]+\left[\frac{\partial \mathbf{F}_v}{\partial \xi}+\frac{\partial \mathbf{G}_v}{\partial \eta}+\frac{\partial \mathbf{H}_v}{\partial \zeta}\right]=\dot{\mathbf{W}}\ . \qquad (3\text{-}11)$$

The delta form of the governing equations is obtained from eq. (3-11) by differentiating with respect to t, taking Jacobians with respect to **U** of the inviscid fluxes ($\mathbf{F}_i,\mathbf{G}_i,\mathbf{H}_i$) and the source term ($\dot{\mathbf{W}}$), and by writing the viscous terms in terms of **U**. Multiplying the result by Δt and expanding the time derivative (∂U/∂t) in a forward difference yields the definitions:

$$\delta \mathbf{U}^{n+1}=\left[\Delta t\frac{\partial \mathbf{U}}{\partial t}\right]^{n+1},$$

$$\Delta \mathbf{U}^{n}=\left[\Delta t\frac{\partial \mathbf{U}}{\partial t}\right]^{n}, \qquad (3\text{-}12)$$

where n is the time level. In these definitions, $\delta \mathbf{U}^{n+1}$ is the implicit delta, and $\Delta \mathbf{U}^{n}$ is the explicit delta.

Applying the definitions in eq. (3-12) to the operations described above, the delta form of the governing equations is obtained:

$$\left\{ \mathbf{I} + \Delta t \left[\frac{\partial}{\partial \xi}\mathbf{A} + \frac{\partial}{\partial \eta}\mathbf{B} + \frac{\partial}{\partial \zeta}\mathbf{C} \right] \right.$$

$$+ \Delta t \left[\frac{\partial}{\partial \xi}\left[\mathbf{F}_v \frac{\partial}{\partial \xi} \right] \mathbf{N} + \frac{\partial}{\partial \eta}\left[\mathbf{G}_v \frac{\partial}{\partial \eta} \right] \mathbf{N} + \frac{\partial}{\partial \zeta}\left[\mathbf{H}_v \frac{\partial}{\partial \zeta} \right] \mathbf{N} \right] \qquad (3\text{-}13)$$

$$\left. - \Delta t \mathbf{D} \right\} \delta \mathbf{U}^{n+1} = \Delta \mathbf{U}^n \ ,$$

where **I** is the identity matrix, $\mathbf{A} = \partial \mathbf{F}_i / \partial \mathbf{U}$, $\mathbf{B} = \partial \mathbf{G}_i / \partial \mathbf{U}$, $\mathbf{C} = \partial \mathbf{H}_i / \partial \mathbf{U}$ are the Jacobians of the inviscid momentum fluxes, $\mathbf{D} = \partial \dot{\mathbf{W}} / \partial \mathbf{U}$ is the source term Jacobian, and **N** is the matrix relating the nonconservative vector $\delta \mathbf{V} = [\delta\rho_1,\ldots,\delta\rho_s,\ldots,\delta\rho_{N_s},\delta u,\delta v,\delta w,\delta T\,]^T$ to the conservative vector $\delta\mathbf{U}$, as $\delta \mathbf{V} = (\partial \mathbf{V}/\partial \mathbf{U}\,)\delta\mathbf{U} = \mathbf{N}\delta\mathbf{U}$. Here, Δt is the time step, and superscript n denotes the time level.

Equation (3-13) represents the delta form of the governing equations for the solution of the Navier-Stokes equations with finite rate chemistry.

Flux Splitting. Solving eq. (3-13) in its present form using finite difference or finite volume techniques would require some form of numerical dissipation to keep the solution stable. Discrete approximations to the Navier-Stokes equations suffer from discretization errors which restrict the stability and robustness of the formulation unless some form of artificial dissipation is added. Numerical dissipation can be provided by "artificial viscosity" formulations, such as the fourth order dissipation function described by Beam and Warming (ref. 17), or by a combination of second- and fourth-order dissipation, such as that proposed by Jameson (ref. 18). The disadvantage of such artificial dissipation functions is that they are relatively empirical, and the constants can be adjusted without regard to the physics of the flowfield. User-tunable dissipation formulations are not desirable for general purpose codes such as HYLDA.

Total variation diminishing (TVD) schemes represent another method for adding artificial dissipation to the governing equations. TVD schemes have been shown by Yee (ref. 19) to be nonoscillatory near strong shocks, but are more computationally intensive, and are not included in the HYLDA code.

An alternative form of dissipation can be obtained by using inviscid flux splitting, as first proposed by Steger and Warming (ref. 10). Initially developed for the Euler equations, flux splitting treats the inviscid terms as combinations of forward and backward facing contributions, depending upon the sign of the eigenvectors. In this manner, flux splitting closely models the physics of the flowfield. Additionally, since the difference schemes are one-sided, the formulation is inherently dissipative (for first order flux splitting), thereby eliminating the need for an arbitrary artificial dissipation term. Flux splitting the inviscid fluxes also increases diagonal dominance of the block tridiagonal matrix obtained from the implicit governing equation, thereby enhancing the stability of the tridiagonal matrix inversion.

Since inviscid flux splitting requires little user-interface to the model, it is the method of choice for the HYLDA code. The flux splitting used in HYLDA is based on the original Steger and Warming (ref. 10) splitting, as modified by MacCormack (ref. 20). A brief overview of the inviscid flux splitting operation is included here.

Flux splitting is added to the implicit (left) side of the governing equations (3-13) by splitting the inviscid flux Jacobians **A, B, C** into contributions because of negative and positive eigenvalues, as follows:

$$\mathbf{A} = \mathbf{A}^+ + \mathbf{A}^- \quad , \tag{3-14}$$

where superscript + denotes positive eigenvalue terms and superscript - denotes negative eigenvalue terms. The inviscid flux Jacobians **B** and **C** are also split as in eq. (3-14).

Flux splitting is implemented in the explicit (right) side of the governing equations (3-13) in a slightly different form. The explicit solution delta ΔU^n includes inviscid terms of the form $\partial F_i/\partial\xi$. Steger and Warming (ref. 10) have shown for an ideal gas that the inviscid terms are homogeneous and of order one, so that $\mathbf{F}_i$ may be written as $\mathbf{F}_i = (\partial \mathbf{F}_i/\partial\xi\,)\mathbf{U}$. The use of a one-temperature gas chemistry model also allows the inviscid flux vectors to retain the homogeneity property required for flux splitting, without requiring the solution of additional energy equations (ref. 21). Since the inviscid fluxes are still homogeneous of order one, the inviscid flux terms on the explicit side of eq. (3-13) may be expanded as:

$$\frac{\partial \mathbf{F}_i}{\partial\xi} = \frac{\partial}{\partial\xi}\mathbf{F}_i = \frac{\partial}{\partial\xi}\left[\frac{\partial \mathbf{F}_i}{\partial\xi}\mathbf{U}\right] = \frac{\partial}{\partial\xi}(\,\mathbf{AU}\,). \tag{3-15}$$

Equation (3-15) is in a form similar to that described above for the implicit side of the governing equations, and again the flux Jacobian may be split into its positive and negative contributions, yielding:

$$\frac{\partial \mathbf{F}_i}{\partial \xi} = \frac{\partial}{\partial \xi}(\mathbf{A}^{+}\mathbf{U} + \mathbf{A}^{-}\mathbf{U}). \tag{3-16}$$

The operations in eq. (3-15) and (3-16) are also repeated for the remaining inviscid fluxes (**B, C**). Substituting the results of eqs. (3-15)-(3-16) into the governing equations (3-13) yields the flux split form of the governing equations:

$$\Bigg\{ \mathbf{I} + \Delta t\left[\frac{\partial}{\partial \xi}\mathbf{A}^{+} + \frac{\partial}{\partial \xi}\mathbf{A}^{-} + \frac{\partial}{\partial \eta}\mathbf{B}^{+} + \frac{\partial}{\partial \eta}\mathbf{B}^{-} + \frac{\partial}{\partial \zeta}\mathbf{C}^{+} + \frac{\partial}{\partial \zeta}\mathbf{C}^{-} + \right]$$

$$+ \Delta t\Bigg[\frac{\partial}{\partial \xi}\left[\mathbf{F}_{\xi}\frac{\partial}{\partial \xi}\right]\mathbf{N} + \frac{\partial}{\partial \eta}\left[\mathbf{G}_{\eta}\frac{\partial}{\partial \eta}\right]\mathbf{N} + \frac{\partial}{\partial \zeta}\left[\mathbf{H}_{\zeta}\frac{\partial}{\partial \zeta}\right]\mathbf{N}\Bigg] - \Delta t\,\mathbf{D}\Bigg]\Bigg\}\delta\mathbf{U}^{n+1}$$

$$= -\Delta t\Bigg\{ \frac{\partial}{\partial \xi}(\mathbf{A}^{+}\mathbf{U}) + \frac{\partial}{\partial \xi}(\mathbf{A}^{-}\mathbf{U}) + \frac{\partial \mathbf{F}_v}{\partial \xi} + \frac{\partial}{\partial \eta}(\mathbf{B}^{+}\mathbf{U}) + \frac{\partial}{\partial \eta}(\mathbf{B}^{-}\mathbf{U}) + \frac{\partial \mathbf{G}_v}{\partial \eta}$$

$$+ \frac{\partial}{\partial \zeta}(\mathbf{C}^{+}\mathbf{U}) + \frac{\partial}{\partial \zeta}(\mathbf{C}^{-}\mathbf{U}) + \frac{\partial \mathbf{H}_v}{\partial \zeta} - \dot{\mathbf{W}}\Bigg\} , \tag{3-17}$$

where inviscid fluxes with positive eigenvalues will be backward differenced, inviscid fluxes with negative eigenvalues will be forward differenced, and viscous terms are central differenced. Note that in eq. (3-17), only diagonal viscous terms are included on the implicit (left) side of the equation, while the explicit (right) side contains full explicit terms. Since the explicit side drives the solution, and the implicit side acts only as a convergence accelerator, the final governing equations exhibit the character of the full Navier-Stokes equations with fully-coupled finite rate chemistry.

The HYLDA code uses two types of flux splitting - the MacCormack weak form and the Steger/Warming strong form. The two formulations differ only in the indexing selected to discretize the inviscid fluxes. The MacCormack formulation is less dissipative than the Steger/Warming form, and is applied to all flow regions away from strong shocks and other large pressure gradient regions. For large pressure gradient regions, the MacCormack form does not provide enough dissipation to keep the solution stable, and the stronger Steger/Warming form is applied instead. Switching between the two methods is triggered by the pressure gradient.

The indexing used by each method is illustrated in figure 2 and table 4 below. Figure 2 shows a stencil of two computational cells in the i-direction. Positive inviscid fluxes travel from left to right in the figure, while negative fluxes travel right to left. The discretized form of the governing equations solves for the flux across the cell face (cell i + 1/2).

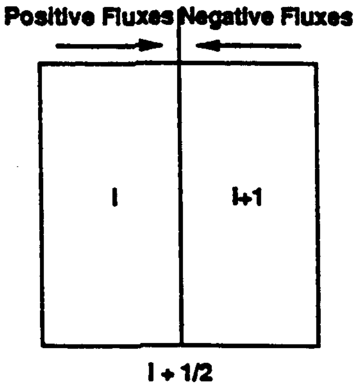


Figure 2. Flux Splitting Stencil

Table 4 shows the indexing schemes for both the MacCormack and the Steger/Warming formulations:

Table 4. Flux Splitting Indexing

	Positive fluxes		Negative fluxes	
Method	Implicit	Explicit	Implicit	Explicit
MacCormack	$A^{+}_{i+½}$	$A^{+}_{i+½}U_i$	$A^{-}_{i+½}$	$A^{-}_{i+½}U_{i+1}$
Steger/Warming	A^{+}_{i}	$A^{+}_{i}U_i$	A^{-}_{i+1}	$A^{-}_{i+1}U_{i+1}$

Similar operations apply to the inviscid fluxes in the other two directions.

These flux splitting indices are used in the discretized governing equations, to be discussed next.

Solution Procedure. The solution procedure for the governing equations (3-17) consists of discretization of the governing equations, and solution of the discretized equations. The flux split governing equation (3-17) is discretized using a finite volume approach, with negative eigenvalue inviscid fluxes being forward differenced, positive eigenvalue inviscid fluxes backward differenced, and viscous terms central differenced. Indices for the inviscid terms vary according to whether MacCormack weak flux splitting or Steger/Warming strong flux splitting is employed, subject to the rules discussed in the previous section.

The discretized form of equation (3-17) represents the fully-implicit governing equation, and as such is a nonlinear equation for which practical solution methods are not available. To make the governing equations tenable, two operations must be performed. First, the nonlinear terms are linearized to the previous time levels, resulting in a banded block septadiagonal matrix, the solution of which is still not currently practical. Next, the difference equation is cast into block tridiagonal form, for which current solution techniques are practical, by selecting the j-direction as the implicit direction. The deltas in the i- and k-directions are relaxed to the previous time level and taken to the right-hand side, resulting in a block tridiagonal algorithm of the form:

$$\mathbf{a}_1^{n+1}\delta\mathbf{U}^{n+1}_{i,j+1,k} + \mathbf{a}_0^{n+1}\delta\mathbf{U}^{n+1}_{i,j,k} + \mathbf{a}_2^{n+1}\delta\mathbf{U}^{n+1}_{i,j-1,k} =$$

$$\Delta\mathbf{U}^{n}_{i,j,k} - \mathbf{a}_3^{n}\delta\mathbf{U}^{n}_{i+1,j,k} - \mathbf{a}_4^{n}\delta\mathbf{U}^{n}_{i-1,j,k} - \mathbf{a}_5^{n}\delta\mathbf{U}^{n}_{i,j,k+1} + \mathbf{a}_6^{n}\delta\mathbf{U}^{n}_{i,j,k-1} \;, \qquad (3\text{-}18)$$

where like terms have been collected into the following definitions:

$$\mathbf{a}_0^{n+1} = \mathbf{I} + \Delta t\left\{ -\left[\mathbf{A}^-_{i-m,j,k} + \left(\mathbf{F}_\xi\mathbf{N} \right)_{i-½,j,k} \right] + \left[\mathbf{A}^+_{i+m,j,k} + \left(\mathbf{F}_\xi\mathbf{N} \right)_{i+½,j,k} \right] \right.$$

$$+ \left[\mathbf{B}^-_{i,j-m,k} + \left(\mathbf{G}_\eta\mathbf{N} \right)_{i,j-½,k} \right] + \left[\mathbf{B}^+_{i,j+m,k} + \left(\mathbf{G}_\eta\mathbf{N} \right)_{i,j+½,k} \right]$$

$$+ \left[\mathbf{C}^-_{i,j,k-m} + \left(\mathbf{H}_\zeta\mathbf{N} \right)_{i,j,k-½} \right] + \left[\mathbf{C}^+_{i,j,k+m} + \left(\mathbf{H}_\zeta\mathbf{N} \right)_{i,j,k+½} \right]$$

$$\left. - \mathbf{D}_{i,j,k} \right\} \;,$$

$$a_1^{n+1} = \Delta t \left\{ B^-_{i,j+m,k} + \left[G_\eta N \right]_{i,j+½,k} \right\} ,$$

$$a_2^{n+1} = \Delta t \left\{ -B^+_{i,j-m,k} + \left[G_\eta N \right]_{i,j-½,k} \right\} ,$$

$$a_3^{n} = \Delta t \left\{ A^-_{i+m,j,k} + \left[F_\xi N \right]_{i+½,j,k} \right\} ,$$

$$a_4^{n} = \Delta t \left\{ -A^+_{i-m,j,k} + \left[F_\xi N \right]_{i-½,j,k} \right\} ,$$

$$a_5^{n} = \Delta t \left\{ C^-_{i,j,k+m} + \left[H_\zeta N \right]_{i,j,k+½} \right\} ,$$

$$a_6^{n} = \Delta t \left\{ -C^+_{i,j,k-m} + \left[H_\zeta N \right]_{i,j,k-½} \right\} . \qquad (3\text{-}19)$$

Here, the subscript m depends upon the type of inviscid flux splitting employed, as defined in the previous section, with $m = 1$ for Steger/Warming strong flux splitting, and $m = ½$ for MacCormack weak flux splitting.

Equation (3-18), with the definitions in eq. (3-19), represent the discretized governing equations for the fully-coupled Navier-Stokes/finite rate chemistry equations.

The solution technique selected for the HYLDA code is based on the MacCormack implicit block tridiagonal method using Gauss-Seidel line relaxation in the non-implicit directions (ref. 9). The solution is updated along implicit lines using standard block tridiagonal inversion methods. The remaining two directions are updated by successive solution sweeps through the computational domain. In principle, the iteration sweeps should be continued until the deltas in the sweep directions converge. In practice, 2-3 sweeps per iteration are sufficient unless time accurate solutions are required.

The solution process follows a one-step implicit update, as follows:

a. Calculate the explicit delta:

$$\Delta U^n_{i,j,k} = -\Delta t \left\{ \frac{D}{\Delta\xi}F + \frac{D}{\Delta\eta}G + \frac{D}{\Delta\zeta}H \right\}^n_{i,j,k} \quad , \qquad (3\text{-}20)$$

where the operators $\frac{D}{\Delta\xi}$, $\frac{D}{\Delta\eta}$, and $\frac{D}{\Delta\zeta}$ represent the discretization and flux splitting operations described in the previous sections.

b. Calculate the implicit delta for $\delta U^{n+1}_{i,j,k}$:

$$\left\{ I + \ldots \right\} \delta U^{n+1}_{i,j,k} = \Delta U^n_{i,j,k} \quad ; \qquad (3\text{-}21)$$

c. Update the solution:

$$U^{n+1}_{i,j,k} = U^n_{i,j,k} + \delta U^{n+1}_{i,j,k} \quad . \qquad (3\text{-}22)$$

The block tridiagonal solver used here is capable of solving an arbitrary size block tridiagonal system of equations. Thus, no change in the solver is required to change from an ideal gas to a multiple species reacting gas case, in keeping with the desire to provide a flexible code structure.

3.3 Boundary Conditions

The HYLDA code boundary condition treatments are implemented implicitly to maintain code stability. The following boundary conditions are available:

- Supersonic inflow;
- Supersonic outflow;
- Supersonic freestream;
- Solid wall (no-slip);
  - Adiabatic or specified temperature;
  - Noncatalytic or catalytic;

- Laminar or turbulent;

- Solid wall (free-slip);
- Symmetry.

The implicit boundary conditions are derived from known explicit boundary conditions by the following procedure:

a. Specify the known explicit boundary condition in terms of the nonconservative solution vector **V**:

$$V_{bc} = Z'V_{adj} + \Gamma \quad , \tag{3-23}$$

where $V = \left[\rho_1,\ldots,\rho_s,\ldots,\rho_{N_s},u,v,w,T \right]^T$, subscript bc denotes the boundary cell, and subscript adj denotes the cell across the boundary surface (adjacent) to the boundary cell.

b. Develop the implicit boundary condition from (a) by deriving the delta form of the boundary condition, as follows:

$$\delta V_{bc} = Z'\delta V_{adj} + V_{adj}\delta Z' + \delta\Gamma \quad . \tag{3-24}$$

Now, δV may be written in terms of the implicit solution vector delta δU as:

$$\delta V = \left[\frac{\partial V}{\partial U} \right] \delta U = N\delta U \quad . \tag{3-25}$$

Substituting eq. (3-25) into eq. (3-24) and solving for δU yields:

$$\delta U_{bc} = \left[N_{bc}^{-1} Z' N_{adj} \right] \delta U_{adj} + \left[N_{bc}^{-1} V_{adj} \delta Z' + N_{bc}^{-1} \delta \Gamma \right] \tag{3-26}$$

$$= Z \delta U_{adj} + \chi \ .$$

For the boundary conditions considered here, $\chi = 0$, and all of the implicit boundary conditions currently contained in the HYLDA code may be represented as:

$$\delta U_{bc} = Z \delta U_{adj} \ . \tag{3-27}$$

The remainder of this section describes the explicit boundary conditions used in the code. The implicit boundary conditions have all been derived according to the development of eq. (3-27) above.

Supersonic Inflow, Freestream. The supersonic inflow and freestream conditions maintain the user-specified freestream conditions, as:

$$V_{bc} = V_{\infty} \ , \tag{3-28}$$

where subscript ∞ denotes user-specified freestream values.

Supersonic Outflow. The supersonic outflow boundary condition extrapolates the flow in the adjacent cell to the boundary cell, as:

$$V_{bc} = V_{adj} \ . \tag{3-29}$$

Solid Wall (No-slip). Although the exact form of the solid wall (no-slip) boundary condition depends on the options selected by the user, the explicit boundary condition can be represented in the general form:

$$\begin{bmatrix} \rho_1 \\ \cdot \\ \cdot \\ \cdot \\ \rho_s \\ \cdot \\ \cdot \\ \cdot \\ \rho_{N_s} \\ u \\ v \\ w \\ T \end{bmatrix}_{bc} = \begin{bmatrix} 2\rho_{1_{wall}} - \rho_{1_{adj}} \\ \cdot \\ \cdot \\ \cdot \\ 2\rho_{s_{wall}} - \rho_{s_{adj}} \\ \cdot \\ \cdot \\ \cdot \\ 2\rho_{N_{s_{wall}}} - \rho_{N_{s_{adj}}} \\ u_{adj} \\ v_{adj} \\ w_{adj} \\ 2T_{wall} - T_{adj} \end{bmatrix} , \qquad (3\text{-}30)$$

where

$\rho_{s_{wall}} = \rho_{s_{adj}}$ if noncatalytic wall

= calculated if catalytic wall

and

$T_{wall} = T_{adj}$ if adiabatic wall

= user–specified constant if isothermal wall

The catalytic wall boundary conditions are discussed below.

Catalytic Wall. In the nonequilibrium flow over a body, the solid surface may act as a catalyst for the recombination of atoms and ions: hence, the heat transfer at the surface will increase. Reentry heating data derived from space shuttle missions clearly showed the significance of nonequilibrium gas chemistry on aerodynamic heating in a high-velocity, low-density flight regime. As altitude and/or velocity is increased above shuttle reentry levels, nonequilibrium effects become even more pronounced. Consequently, the capability to model surface kinetics is essential in developing a flowfield code applicable to high-speed, low-density flows.

The HYLDA code models wall catalyticity effects as part of the solid wall, no-slip boundary condition. Wall catalycity is represented in terms of species catalytic efficiencies, defined as the ratio of the number of species recombining at the wall to the number of species arriving at the wall. Determination of the species catalytic efficiency is dependent upon experimental measurements, and there is a degree of uncertainty as to the best possible way to represent surface catalytic effects.

The HYLDA code implements the catalytic boundary condition in a lagged manner. That is, the new values of the species densities at the wall because of catalycity are calculated at the completion of each computational step, rather than as part of the step. The governing equation for the calculation of the catalytic wall species densities is:

$$\left(\rho\hat{q}_s\right)_{wall} - \delta_s\left(\frac{\rho_s\gamma_s}{2-\gamma_s}\right)_{wall}\left(\frac{R_sT_{wall}}{2\pi}\right)^{1/2}, \qquad (3\text{-}31)$$

where

$\rho =$ total density,

$\hat{q}_s =$ species s diffusion velocity, defined in eq. (3-4),

$\gamma_s =$ species s catalytic efficiency,

$\delta_s =$ +1 for polyatomic species, -1 for monatomic species,

$R_s = \dfrac{\hat{R}}{M_s}$,

with $\hat{R}$ being the Universal Gas Constant and M_s being the species molecular weight.

Equation (3-31) was derived subject to the following conditions:

- No blowing, ablation, or porous surface;
- Neglects other species (i.e., assumes zero concentration for all species except the species being calculated);

- Uses the Lagmuir-Hinschelwood recombination mechanism;
- Assumes first order surface catalytic recombination.

Applying the definition of the diffusion velocity at the wall, the left-hand-side of eq. (3-31) becomes:

$$\left[\rho\hat{q}_s\right]_{wall} = -\left[\rho D_{s_{mix}} \frac{M_s}{M_{mix}} \frac{\partial X_s}{\partial \hat{n}}\right]_{wall} , \tag{3-32}$$

where $D_{s_{mix}}$ is the mixture diffusion coefficient for species s (defined in section 4.0), M_s and M_{mix} are the species and mixture molecular weights, respectively, and $\partial X_s/\partial\hat{n}$ is the gradient of the species mole fraction, with $X_s = (\rho_s/M_s)/(\rho/M_{mix})$.

Substituting eq. (3-32) into eq. (3-31) and solving for $\partial X_s/\partial\hat{n}$ yields:

$$\left[\frac{\partial X_s}{\partial \hat{n}}\right]_{wall} = \left[\frac{-X_s}{D_{s_{mix}}}\right]_{wall} \left[\frac{\gamma_s \delta_s}{2-\gamma_s}\right] \left[\frac{R_s T_{wall}}{2\pi}\right]^{½} . \tag{3-33}$$

Expanding the gradient term on the left-hand-side of eq. (3-33) in a first order difference and solving for the molar concentration in the boundary cell yields:

$$\frac{\partial X_s}{\partial \hat{n}} = \frac{\left[X_{s_{adj}} - X_{s_{bc}}\right]}{\Delta\hat{n}} ,$$

$$X_{s_{bc}} = X_{s_{adj}} - \Delta\hat{n}\left[\frac{\partial X_s}{\partial \hat{n}}\right]_{wall} , \tag{3-34}$$

where $\left[\frac{\partial X_s}{\partial \hat{n}}\right]_{wall}$ is given by eq. (3-33).

Finally, the boundary cell species density is obtained from:

$$\rho_{s_{bc}} = X_{s_{bc}} \frac{\rho_{bc}}{M_{mix_{bc}}} M_s \quad . \tag{3-35}$$

Turbulent Wall. Hypersonic, high-altitude, low-density, low-Reynolds number flows such as those of interest to this contract are primarily laminar flows. In the late phases of reentry, though, transition to turbulence may occur. The turbulence expected for the cases of interest here include shear layer type turbulence on the windward side of the reentry vehicle, as well as vortex-separated flows on the leeside of the vehicle.

Windward turbulence of the shear layer type is expected to occur as a shock/boundary layer interaction, such as might be found at a deflected control surface for example. This shear layer type turbulence is relatively simple to model, even with basic algebraic turbulence models such as the Baldwin-Lomax model (ref. 4).

Leeside turbulence occurs when the vehicle angle of attack is sufficient to separate the flow over the leeside surfaces, forming vortices that can strongly influence heating patterns. These leeside flows are highly sensitive to vehicle geometry, angle of attack, and Reynolds number.

The Interim Report for Phase I of this contract (ref. 2) discussed the turbulence model selection process in the context of both shear flows and leeside flows. The turbulence model selected was the Baldwin-Lomax algebraic turbulence model (ref. 4) subject to the modifications of Degani-Schiff (ref. 22), and also including a wall function (law-of-the-wall) formulation. Details of the model are included here.

The selection of the Baldwin-Lomax model over more complex models was in part influenced by the need to calculate finite rate chemistry with its inherently large number of equations. The increased accuracy obtained by more complex models was not considered sufficient to offset the increased computing time required for higher-order models, especially when considering finite rate chemistry calculations with large numbers of species. The HYLDA code is, however, configured to permit incorporation of higher-order turbulence models.

The Baldwin-Lomax model calculates turbulent viscosity according to:

$$\mu_{turb} = \left[\mu_{turb} \right]_{inner} \text{ if } y<y_{cross}$$

$$= \left[\mu_{turb} \right]_{outer} \text{ if } y>y_{cross} \qquad (3\text{-}36)$$

where

$$\left[\mu_{turb} \right]_{inner} = \rho\kappa^2 h^2 F_{VD}^2 \,|\,\omega\,| \quad ,$$

$$\left[\mu_{turb} \right]_{outer} = \kappa C_{cp} F_{wake} F_{KLEB} \quad , \qquad (3\text{-}37)$$

and y_{cross} is the smallest distance normal to the solid wall at which the inner and outer turbulence viscosities are equal.

The variables in eq. (3-37) are defined as follows:

ρ = density,
κ = von Karman constatnt (=0.41)
h = distance normal the wall
ω = vorticity
C_{cp} = constant (=1.6)

and the Van Driest (F_{VD}) and Klebanoff (F_{KLEB}) functions are defined as:

$$F_{VD} = \left[1 - \exp\left(-y^+/A^+ \right) \right] \quad ,$$

$$F_{KLEB} = \left[1 + 5.5 \left(\frac{C_{KLEB}\,|\,h\,|}{|\,\hat{h}\,|} \right)^6 \right]^{-1} \quad . \qquad (3\text{-}38)$$

The variables in eq. (3-38) are defined as follows:

$$y^+ = \frac{u_\tau y \rho_{wall}}{\mu_{wall}} \quad ,$$

u_τ = friction velocity = $(\tau_{wall}/\rho_{wall})^{1/2}$,

μ_{wall} = wall viscosity ,

τ_{wall} = wall shear stress ,

A^+ = constant (= 26) ,

C_{KLEB} = Klebanoff constant (= 0.3) ,

$\hat{h}$ = the value of h at $\left\{ |h| \, |\omega| F_{VD} \right\}_{max}$.

The wake function F_{wake} is the minimum of two functions:

$$F_{wake} = \min\left[F_{w1}, F_{w2} \right] , \qquad (3\text{-}39)$$

where

$$F_{w1} = |\hat{h}| \left\{ |h| \, |\omega| F_{VD} \right\}_{max} ,$$

$$F_{w2} = \frac{C_{wk} |\hat{h}| q^2_{max}}{\left\{ |h| \, |\omega| F_{VD} \right\}_{max}} , \qquad (3\text{-}40)$$

with C_{wk} = 0.25, and q_{max} equal to the maximum velocity in the boundary layer.

The Baldwin-Lomax model as shown in eqs. (3-36)-(3-40) is sufficient for the wall-bounded turbu!^nt shear layers found on the windward side of hypersonic vehicles. As discussed in the Phase I Interim Technical Report (ref. 2), however, the definition of the length scale $\hat{h}$ in eq. (3-38) leads to an improper length scale determination when calculating leeside flows. To correct the deficiency, the Degani-Schiff modification (ref. 22) to choose the first maximum of the function determining $\hat{h}$ in eq. (3-38) has been incorporated.

<u>Wall Functions (Law-of-the-Wall).</u> Turbulence length scales are generally orders of magnitude smaller than laminar length scales, and to adequately resolve a turbulent boundary layer (i.e. grid to within $y^+ < 5$) requires a large number of grid points, and a large amount of

computer time. Wall functions such as the law-of-the-wall formulation used in the HYLDA code are used to relax the amount of grid necessary to provide the correct wall value of viscosity. The law-of-the-wall model predicts the wall viscosity using empirical correlations from experimental studies of turbulent boundary layers. These correlations are applied at greater distances from the wall (typically $y^{+} \sim$ of 100), thereby reducing the number of grid points needed to adequately resolve the turbulent boundary layer.

The governing equation is based on the following expression for the shear velocity (u_τ):

$$u_\tau = \frac{\kappa u^{*}}{\ln\left[e^{\kappa c}\, e^{\pi W} \left[\frac{y\rho_{wall}}{\mu_{wall}} \right] u_\tau \right]} \qquad (3\text{-}41)$$

where

u_τ = shear velocity (defined in eq.(3–38)) ,
κ = von Karman constant (= 0.41) ,
$C = 5.1$,
$\pi = 1.25\kappa = 0.5125$ for flat plates ,
W = Cole's tabulated universal wake function (taken here as 0) ,

$$u^{*} = \frac{q_e}{\sigma}\sin^{-1}\left[\frac{\sigma q}{q_e} \right] ,$$

with

$$\sigma = \left[\frac{0.5\,(\gamma - 1\,)\,M_\infty^2}{1 + 0.5\,(\gamma - 1\,)\,M_\infty^2} \right] .$$

Equation (3-41) is iterated to obtain the correct shear velocity at the wall, and the wall viscosity is then extracted from this value as follows:

$$\mu_{\text{wall}} = \frac{\rho_{\text{wall}} u_{\tau}^{2}}{\left[\dfrac{\partial u}{\partial \hat{n}}\right]_{\text{wall}}} \quad . \tag{3-42}$$

Solid Wall (Free-slip), Symmetry. The solid wall (free-slip) and symmetry boundary conditions ensure that all fluxes are reflected at the boundary:

$$\mathbf{V}_{\text{bc}} = \mathbf{R}\mathbf{V}_{\text{adj}} \quad , \tag{3-43}$$

where **R** is the rotation matrix, defined as:

$$\mathbf{R} = \begin{bmatrix}
1 & \cdots & 0 & \cdots & 0 & 0 & 0 & 0 & 0 \\
\vdots & & \vdots & & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & \cdots & 1 & \cdots & 0 & 0 & 0 & 0 & 0 \\
\vdots & & \vdots & & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & \cdots & 0 & \cdots & 1 & 0 & 0 & 0 & 0 \\
0 & \cdots & 0 & \cdots & 0 & 1-2l_{m1}l_{m1} & -2l_{m1}l_{m2} & -2l_{m1}l_{m3} & 0 \\
0 & \cdots & 0 & \cdots & 0 & -2l_{m2}l_{m1} & 1-2l_{m2}l_{m2} & -2l_{m2}l_{m3} & 0 \\
0 & \cdots & 0 & \cdots & 0 & -2l_{m3}l_{m1} & -2l_{m3}l_{m2} & 1-2l_{m3}l_{m3} & 0 \\
0 & \cdots & 0 & \cdots & 0 & 0 & 0 & 0 & 1
\end{bmatrix} \tag{3-44}$$

with l_{mj} being the direction cosines.

4.0 THERMOCHEMISTRY MODELS

The HYLDA code contains three types of thermochemistry models: ideal gas, equilibrium air, and finite rate (nonequilibrium) air. In general, thermochemical models provide the chemistry source terms, thermodynamic properties, and transport properties for the calculation scheme. For ideal gases and equilibrium air the source terms are zero. For an ideal gas, the thermodynamic pressure and temperature are calculated directly from the ideal gas relationships derived from eqs. (3-5) and (3-8) respectively. The equilibrium air model is empirical, and provides pressure, temperature, speed of sound, and ratio of specific heats (γ) given internal energy and density using Tannehill's real gas model (ref. 5). For both ideal gases and equilibrium air, the transport property viscosity is obtained using Sutherland's Law, and the thermal conductivity is determined using a viscosity and Prandtl number relationship.

For finite rate chemistry flows, the source terms are nonzero, the thermodynamic and transport properties depend upon the properties of the constituent species. Finite rate chemistry flows require detailed models to account for nonequilibrium effects on the source terms, thermodynamics, and transport properties. The results of this contract effort include the development of an air chemistry model, as well as a transport properties model and species thermodynamics model. The remainder of this sections details these models.

4.1 Air Chemistry Model

An extensive literature search was performed for chemical and thermal nonequilibrium experimental data. Several sources of shock tube experiments were found (refs. 23, 24) which could be used to calibrate air nonequilibrium chemistry mechanisms. The data were considered sufficiently accurate to compare various chemical nonequilibrium models (i.e., one-temperature models). However, the same was not true for thermal nonequilibrium (multi-temperature) experimental data sets. Very few data sets were found which contained useful measurements of thermal nonequilibrium processes. Based on the results of the literature search we decided that only one-temperature air chemistry models could be calibrated to engineering accuracy and would be considered in this contract. Thus, rotation, translation, and vibration energy modes were assumed to be in equilibrium.

Many air ionization models were identified, with each model utilizing different chemical species, reaction paths, rate coefficients, and application features. For instance, a model may depend on one (translation) or two (translation and vibration) temperatures, may have up to 15 chemical species, may use an Arrhenius or non-Arrhenius reaction rate, and may have as many as 50-60 reaction paths. These variations make a thorough analysis extremely tedious and difficult. To insure a feasible analysis, the following assumptions and restrictions were employed:

- Only Arrhenius reaction rates were considered.
- Chemical reaction rates were assumed to depend only on the translational temperature (T); i.e., one-temperature reaction rates.
- Chemical reaction rates were assumed to be valid up to 17000 K.
- Argon has a negligible effect in high-temperature air reactions.

A variety of species have been employed in these air chemistry models; however, the species that can be utilized in any calculation is also dependent on the available species thermodynamic and transport properties. A review of species thermodynamic and transport properties at the initiation of this contract concluded that at least nine air species (N_2,O_2,N,O,NO^+,N^+,O^+,E^-) could be modeled. These species appear to adequately model high-temperature air in most engineering problems; hence, the thermochemistry capabilities curren'ly included in HYLDA employ the nine species mentioned above. HYLDA can easily accommodate additional species if atomic/molecular information for each species is available.

Of the many models examined, five air ionization models utilized an Arrhenius reaction model of the form $A_i T^{N_i} \exp(-E_i/T)$ where A_i,N_i,E_i are rate coefficients. These models were developed by K. Wray, C. Park and G.Menees, S. Kang and M. Dunn, D. Bittker, and M. Bortner (refs. 6, 25-28). The Wray model is well known and represents one of the simplest (18 reaction paths, 7 species) ionization models conceived. The Park/Menees model was derived to investigate meteoroid wakes and both the Kang/Dunn and Bortner models were employed for flowfield analyses about reentry vehicles. Finally, the Bittker model was employed to analyze shock-tube air reactions. Note that the Wray model has been modified from its original form. Modifications included the elimination of $N_2 + O_2 = 2$ NO, based on Camac and Feinberg's recommendation (ref. 23) and the assumption that Argon has a negligible effect in high temperature air reactions. The Park/Menees model (ref. 6) has been abbreviated to include only nine air species and is the finite-rate chemistry model included in HYLDA. The Appendix lists the reaction paths and rate constants for the Park/Menees model. For details of the chemistry model selection process, refer to the Phase I Interim Technical Report (ref. 2).

4.2 Thermodynamic Properties

Species thermodynamic properties such as entropy, Gibbs free energy, enthalpy, and specific heat capacity are important components to model development. These properties must be accurately known up to very high temperatures (20000 K). McBride (ref. 8) derived polynomial relationships for the thermodynamic properties of the nine air species considered above. McBride utilized the following relationships for the specific heat capacity (C_p), enthalpy (H) and Gibbs free energy (F):

$$\frac{C_p}{R} = Z_1 + Z_2T + Z_3T^2 + Z_4T^3 + Z_5T^4 \tag{4-1}$$

$$\frac{H}{RT} = Z_1 + Z_2T/2 + Z_3T^2/3 + Z_4T^3/4 + Z_5T^4/5 + Z_6/T \tag{4-2}$$

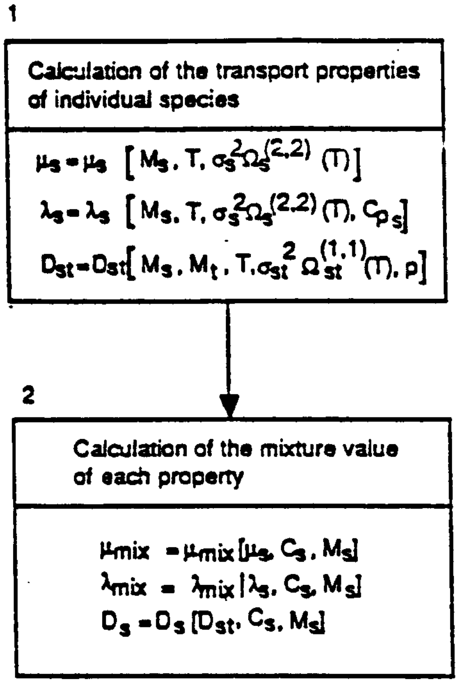
$$\frac{F}{RT} = Z_1[1.0-\log_e(T)] - Z_2T/2 - Z_3T^2/6 - Z_4T^3/12 - Z_5T^4/20 + Z_6/T - Z_7 \tag{4-3}$$

The coefficients for eqs. (4-1) through (4-3) for all nine air species can be found in the Appendix.

4.3 Transport Properties

The transport of fluid momentum, thermal energy, and mass is characterized by the coefficient of viscosity (μ), thermal conductivity (λ), and (concentration) diffusion D, respectively. Viscosity affects the growth of the boundary layer, hence, will influence the distance (and time) particles and energy must travel to reach the wall surface. Conversely, thermal conductivity regulates the transfer of heat, and thus, influences particle motion (i.e., species concentration and thermal diffusion).

All nonequilibrium chemistry methods require the computation of each transport property for individual species in the reactive flowfield. A formula is then used to obtain the mixture value of the property. This procedure is depicted in figure 3, which also shows the functional dependence of each property on other parameters.


1
Calculation of the transport properties of individual species
$\mu_s = \mu_s\ [M_s, T, \sigma_s^2\Omega_s^{(2,2)}(T)]$
$\lambda_s = \lambda_s\ [M_s, T, \sigma_s^2\Omega_s^{(2,2)}(T), C_{p_s}]$
$D_{st} = D_{st}[M_s, M_t, T, \sigma_{st}^2\,\Omega_{st}^{(1,1)}(T), p]$
2
Calculation of the mixture value of each property
$\mu_{mix} = \mu_{mix}[\mu_s, C_s, M_s]$
$\lambda_{mix} = \lambda_{mix}[\lambda_s, C_s, M_s]$
$D_s = D_s[D_{st}, C_s, M_s]$

where μ = viscosity
λ = thermal conductivity
D = concentration diffusion
s = chemical specie s
mix = mixture value
M = molecular weight
T = temperature
$\sigma^2\Omega(l,m)$ = collision cross-section area
C_p = specific heat capacity at constant pressure
p = pressure
C = concentration

Figure 3. Transport Properties Methodology

First-approximations to the species transport properties ca.ı be represented as follows:

$$\mu_{ij} = \frac{2m_i m_j}{(m_i + m_j)\Delta_{ij}^{(2)}} \quad , \qquad (4\text{-}4)$$

$$\lambda_{ij} = \frac{15}{4}\frac{k_{boltz}}{\Delta_{ij}^{(2)}} \quad , \qquad (4\text{-}5)$$

$$D_{ij} = \frac{k_{boltz}T}{P\Delta_{ij}^{(1)}} \quad , \qquad (4\text{-}6)$$

$$\text{where} \quad \Delta_{ij}^{(q)} = C_q\left[\frac{2m_i m_j}{\pi k_{boltz}T(m_i + m_j)}\right]^{1/2}\pi\Omega_{ij}^{(q,q)} \quad , \qquad (4\text{-}7)$$

$$C_1 = \frac{8}{3} \; , \; C_2 = \frac{16}{5} \qquad (4\text{-}8)$$

where μ, λ, D are the coefficients of viscosity, thermal conductivity and mass diffusion; m is the particle mass, k_{boltz} is the Boltzmann constant, P is the pressure and $\Omega^{(q,q)}$ is the collision cross-section (area) normalized with respect to its rigid-sphere value. Equations (4-4) through (4-8) are based on kinetic theory of dilute gases. Complete derivations are provided in ref. 29.

All species transport properties depend on the collision integral cross-section, defined as the probable collision area between species s and t. The collision cross-section is dependent on species mass, velocity, temperature, collision deflection angle, and a potential force function.

The cross-sections for all possible collision pairs, among the nine air species, have been modeled by Nicolet (ref. 7) using a simple logarithmic polynomial function of temperature. The Appendix contains the coefficients for eq. (4-9) for all non-coulomb type collisions of the form:

$$\pi\Omega_{ij}^{l,m} = A_{ij}^{l,m}\log_e[T/1000] + B_{ij}^{l,m} \qquad (4\text{-}9)$$

Coulomb collisions involving two charged particles have been modeled by Liboff assuming an unscreened Coulomb potential with Debye-length cutoff:

$$\pi\Omega_{ij}^{(1,1)} = \pi\Omega_{ij}^{(2,2)} = \frac{\pi(Z_iZ_je^2)^2}{(kT)^2}\left[\log\left[\frac{kT}{Z_iZ_je^3}\left(\frac{kT}{\pi n_e}\right)^{1/2}\right] - 0.577\right] \qquad (4\text{-}10)$$

Once the individual species transport properties are known, the mixture value for each property may be obtained by the mixture relationships of Brokaw (ref. 29):

$$\mu_{mix} = \sum_{i=1}^{i=N} \left\{ \frac{X_i}{\displaystyle\sum_{j=1}^{j=N} \frac{2m_i X_j}{(m_i + m_j)\mu_{ij}}} \right\} \qquad (4\text{-}11)$$

$$\lambda_{trans,mix} = \sum_{i=1}^{i=N} \left\{ \frac{X_i}{\displaystyle\sum_{j=1}^{j=N} \frac{\alpha_{ij} X_j}{\lambda_{ij}}} \right\} \qquad (4\text{-}12)$$

where $$\alpha_{ij} = 1 + \frac{\left[1 - \frac{m_i}{m_j}\right]\left[.45 - 2.54\frac{m_i}{m_j}\right]}{\left[1 + \frac{m_i}{m_j}\right]^2} \qquad (4\text{-}13)$$

$$\lambda_{int,mix} = \frac{P}{T} \sum_{i=1}^{i=N} \left\{ \frac{X_i \left[\frac{Cp_i}{R_i} - 2.5 \right]}{\displaystyle\sum_{j=1}^{j=N} \frac{X_j}{D_{ij}}} \right\} \qquad (4\text{-}14)$$

Finally, the reciprocal of the mixture diffusion coefficient is approximately equal to the mass average of the reciprocal of the binary diffusion coefficients (refs. 30, 31), as follows:

$$D_{i,mix} = \frac{\displaystyle\sum_{i=1,i \neq j}^{i=N} X_j M_j}{M_{mix} \displaystyle\sum_{i=1,i \neq j}^{i=N} \frac{X_j}{D_{ij}}} \qquad (4\text{-}15)$$

The individual species formulas in eqs. (4-4) through (4-8), collision integral cross-section expressions in eqs. (4-9) and (4-10), and the mixture equations (4-11) through (4-14) represent the relationships used by HYLDA to determine the transport properties.

5.0 CODE DEMONSTRATION

The demonstration cases detailed in this section were calculated as part of the contract requirements for the current HYLDA code development effort. These cases were selected by the Air Force for the purpose of demonstrating the HYLDA code's finite rate chemistry capabilities, and should not be interpreted as an attempt to fully validate the HYLDA code. Full validation of the HYLDA code will require many calculations over a wide range of test conditions, a requirement beyond the scope of the current contract. Ongoing (ref. 32) and future Air Force work will address the HYLDA code validation issue in detail.

Demonstration case 1 (section 5.1) consists of a calculation of the flow about a Mach 15 wedge-flat plate combination, and illustrates the HYLDA code capability to calculate reacting air flowfields. Demonstration case 2 (section 5.2) calculates the reacting air flow about a biconic geometry at Mach 6.9, with emphasis on surface heat flux. Demonstration case 3 (section 5.3) calculates the reacting flow about a hemisphere, with emphasis on the effects of wall catalycity.

While these demonstration cases presented here should not be considered full validation cases, they do serve three purposes: (1) to demonstrate the HYLDA code finite rate chemistry capability, (2) to assess the ability of the HYLDA code to model wind tunnel tests, and (3) to illustrate the need for new hypersonic wind tunnel results capable of being used to validate new codes such as HYLDA.

5.1 Case 1: Mach 15 Wedge-Flat Plate Flow

The first demonstration case was based on the wedge-flat plate experimental studies by Vidal and Stoddard (ref. 33). The objective of the experimental studies was to demonstrate the magnitude of nonequilibrium effects as compared to viscous effects. The geometry selected was a flat plate at zero incidence preceded by a wedge leading edge. Vidal and Stoddard studied both thin (0.5 inches, 0.0127 meters) and thick (4.0 inches, 0.1016 meters) plates, with the thin plate intended to simulate nonreacting gas flows, and the thick plate providing reacting gas data. Since the goal of the HYLDA code is to calculate reacting gas flowfields, the focus of this demonstration case was on the thick wedge geometry.

As noted in the introduction to this section, one purpose of the demonstration cases was to illustrate the need for improved wind tunnel tests of hypersonic flowfields. A major problem associated with wind tunnel tests of the early 1960's was the difficulty in accurately measuring and quantifying flowfield and surface data. These experimental difficulties translate into uncertainties in modeling the flows numerically, since in many cases the input parameters for the computer code (including HYLDA) must be extracted from given experimental data.

Another uncertainty in the experimental results from shock tunnels is the effect that the transient response of the tunnel had on the "steady state" reacting flowfield. The ability of the HYLDA code to accurately model shock tunnel data depends upon the influence of the transient response in the shock tunnel on the measured "steady state" flowfield conditions.

The results presented below for this demonstration case assumed the wedge-flat plate geometry to be in the test section of the wind tunnel so that the oncoming flow could be represented as parallel flow. Comparison of the numerical and experimental results showed discrepancies in the pressure field about the geometry. Investigation of both the numerical results and the experimental results revealed that the test geometry was not located in the test section, but rather in the inlet nozzle portion of the wind tunnel. The HYLDA code currently requires a uniform freestream initial condition, so the radial flow of the experimental analysis could not be replicated.

In light of the discrepancy between model locations (and thus initial conditions), quantitative comparison of the numerical and experimental results is not strictly possible. Qualitative comparisons of the results are possible, though, and the results presented here show that some flow features are relatively insensitive to the radial component of velocity, allowing at least a qualitative assessment of the ability of the HYLDA code to model the reacting gas flow over the wedge-flat plate geometry.

The discussion of results is divided into three parts: geometry, initial conditions, and results and discussion.

Geometry. The geometry used for this case consisted of a 4 inch thick (0.1016 meters) flat plate with a 42.5-degree half-angle wedge leading edge. The geometry is shown in fig. 4:

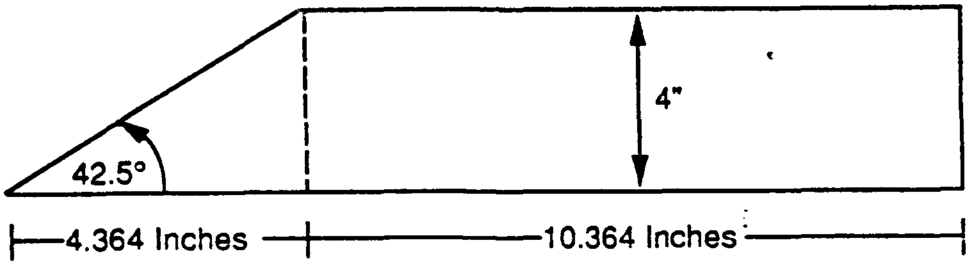


Figure 4. Demonstration Case 1 Geometry

Initial Conditions. Some of the initial conditions required by HYLDA (freestream pressure and temperature) were not explicitly provided by the discussion of the experimental results in ref. 33. These conditions were extracted from the data provided, as follows:

Given: ρ/ρ_{SL}, U_∞, M_∞, and Re/inch;
Extract: P_∞, T_∞

- Determine the freestream density (ρ_∞) from ρ/ρ_{SL}, where ρ_{SL} is the sea level density.
- Determine the freestream viscosity (μ_∞) using ρ_∞, U_∞, and Re/inch.
- Iterate for the freestream temperature (T_∞) using Sutherland's Law and the μ_∞ calculated above:

$$\mu_\infty(T_\infty) = \frac{1.458(10^{-6})T_\infty^{1.5}}{T_\infty + 110.3} \quad ;$$

- Calculate the freestream pressure (P_∞) using the freestream values of density and temperature and the mixture gas constant (for the air chemistry case here, the mixture gas constant was 287.88 J/(kg-K)).

The results below are presented for two of the large-scale wedge cases studied by Vidal and Stoddard, namely, the low stagnation temperature case and the high stagnation temperature case 1 tests. With the procedure outlined above, and using the appropriate parts of table 1 from ref. 33, the initial conditions used here are as follows:

Table 5. Demonstration Case 1 Initial Conditions

	Reservoir conditions*		Ambient at model location**				Additional ambient conditions***	
Case	T_o (R,K)	P_o (psia,N/m^2)	ρ_∞/ρ_{SL}	U_∞ (ft/s,m/s)	Mach #	Re/in,Re/m	P_∞ (psia,N/m^2)	T_∞ (R,K)
Low temp.	5922, 3290	3820, 2.6(10^7)	6.01(10^{-4})	14900, 4542	15.0	9210, 362600	2.1(10^{-3}), 14.43	322, 179
High temp.	11970, 6650	16700, 8.3(10^7)	2.28(10^{-4})	9370, 2856	16.3	2680, 105500	2.6(10^{-3}), 18	153, 85

* calculated from static conditions at model location, data from table 1 (ref. 33)
** calculated from reservoir conditions, data from table 1 (ref. 33)
*** calculated from ref. 33 ambient conditions

Additionally, the geometry wall temperature was assumed to be equal to the ambient pre-test temperature of 300 K (540 R).

R^esults and Discussion. Using the initial conditions in table 5, the HYLDA code was used to calculate the flowfield about the wedge-flat plate geometry for both the low stagnation temperature case and the high stagnation temperature case 1 from the Vidal and Stoddard report (ref. 33). The results and discussions for these cases will be described separately.

Low Stagnation Temperature Results

The grid and the computational domain for the low stagnation temperature case are shown in fig. 5. The grid consists of 80 cells in the streamwise direction, 50 cells in the radial direction, and 1 cell in the spanwise direction. Only one point is required in the third direction since the flow is essentially two-dimensional. The grid is clustered near the nose of the wedge to resolve the shock, and also clustered near the body to resolve the viscous layers.

The low stagnation temperature calculation was accomplished in 112 steps with a maximum CFL number of 500. Figs. 6 through 8 show the Mach, temperature, and γ contours for the flowfield, where γ is the ratio of specific heats. The Mach and temperature contours qualitatively show the expected flow features, including the shock attached to the leading edge of the wedge and the temperature rise through the shock and subsequent cooling to the wall temperature. The γ contours are useful to show where finite rate chemistry effects are largest. As expected, the greatest degree of reacting chemistry occurs in the high temperature region behind the shock along the wedge face.

Vidal and Stoddard (ref. 33) included a Schlieren photograph for this low temperature case. Although the correlation is not exact, density profiles from the numerical calculation can be used to compare to the Schlieren photograph. The comparison of numerically-calculated density contours and experimentally-produced Schlieren photograph is shown in fig. 9(a,b), with part (a) being the Schlieren photograph, and part (b) being the numerical results. The figures show that the HYLDA code predicts the shock wave angles on both the upper and lower surface. The boundary layer on the flat plate is more easily observed in figs. 6 through 8.

As noted in the introduction to this section, direct comparison of quantitative results is not appropriate because of the difference in initial conditions for the experiment and the calculation. The comparison of surface pressures shown in fig. 10, though, demonstrates that the pressure field is not greatly effected by the radial component of velocity in the experimental study.

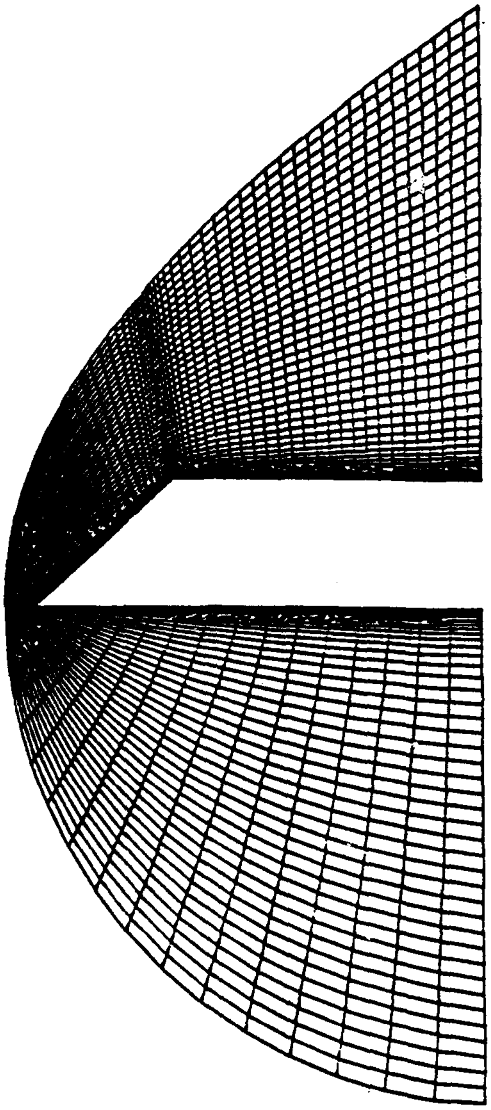
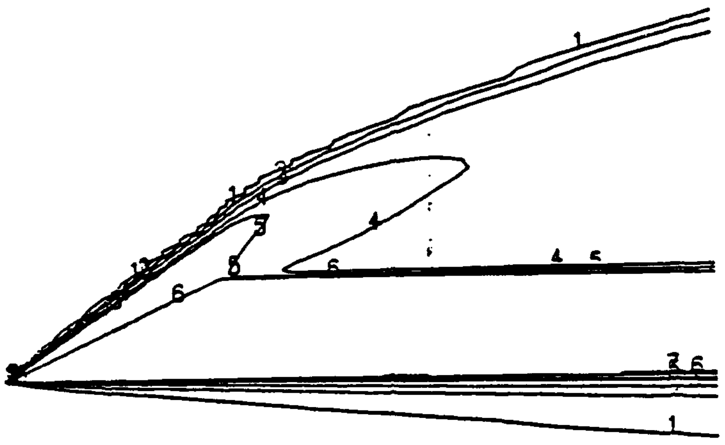
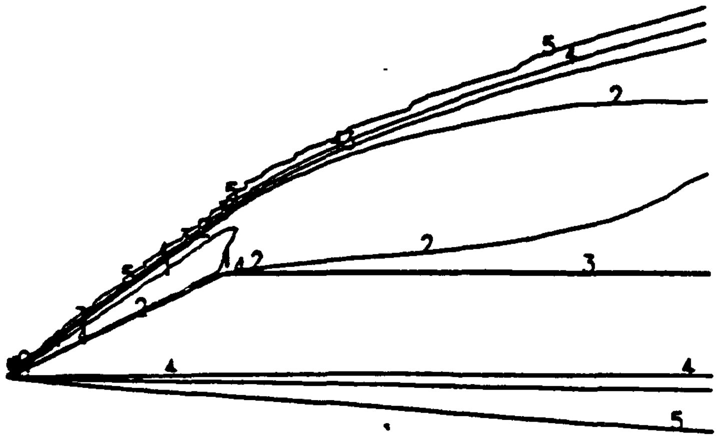


Figure 5. Demonstration Case 1 Mesh for Low Stagnation Temperature Case



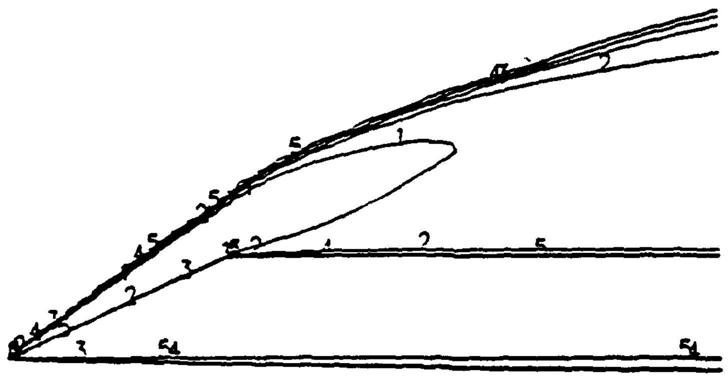
Legend - Mach Number	
1	15.0
2	10.0
3	5.0
4	2.5
5	2.0
6	1.0
7	0.0

Figure 6. Demonstration Case 1 Low Stagnation Temperature Case - Mach Contours



Legend - T / T_∞	
1	12.84
2	8.38
3	5.58
4	2.79
5	0.56
T_∞	179 K

Figure 7. Demonstration Case 1 Low Stagnation Temperature Case - Temperature Contours



Legend - γ	
1	1.30
2	1.32
3	1.34
4	1.36
5	1.38

Figure 8. Demonstration Case 1 Low Stagnation Temperature Case - γ Contours

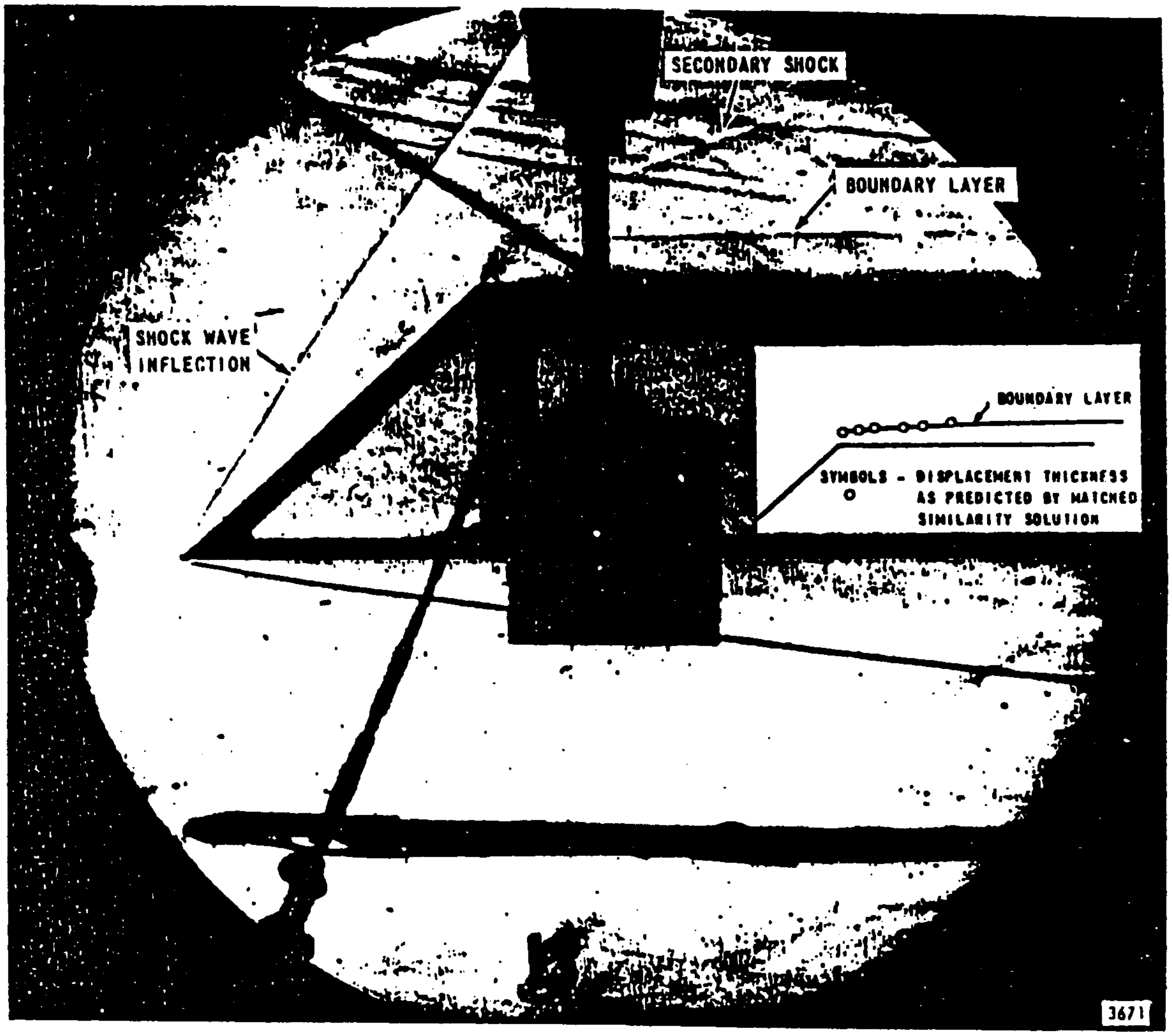

SECONDARY SHOCK
BOUNDARY LAYER
SHOCK WAVE INFLECTION
BOUNDARY LAYER
SYMBOLS - DISPLACEMENT THICKNESS
○ AS PREDICTED BY MATCHED SIMILARITY SOLUTION
3671

Figure 9a. Demonstration Case 1 Low Stagnation Temperature Case - Schlieren (ref. 33)

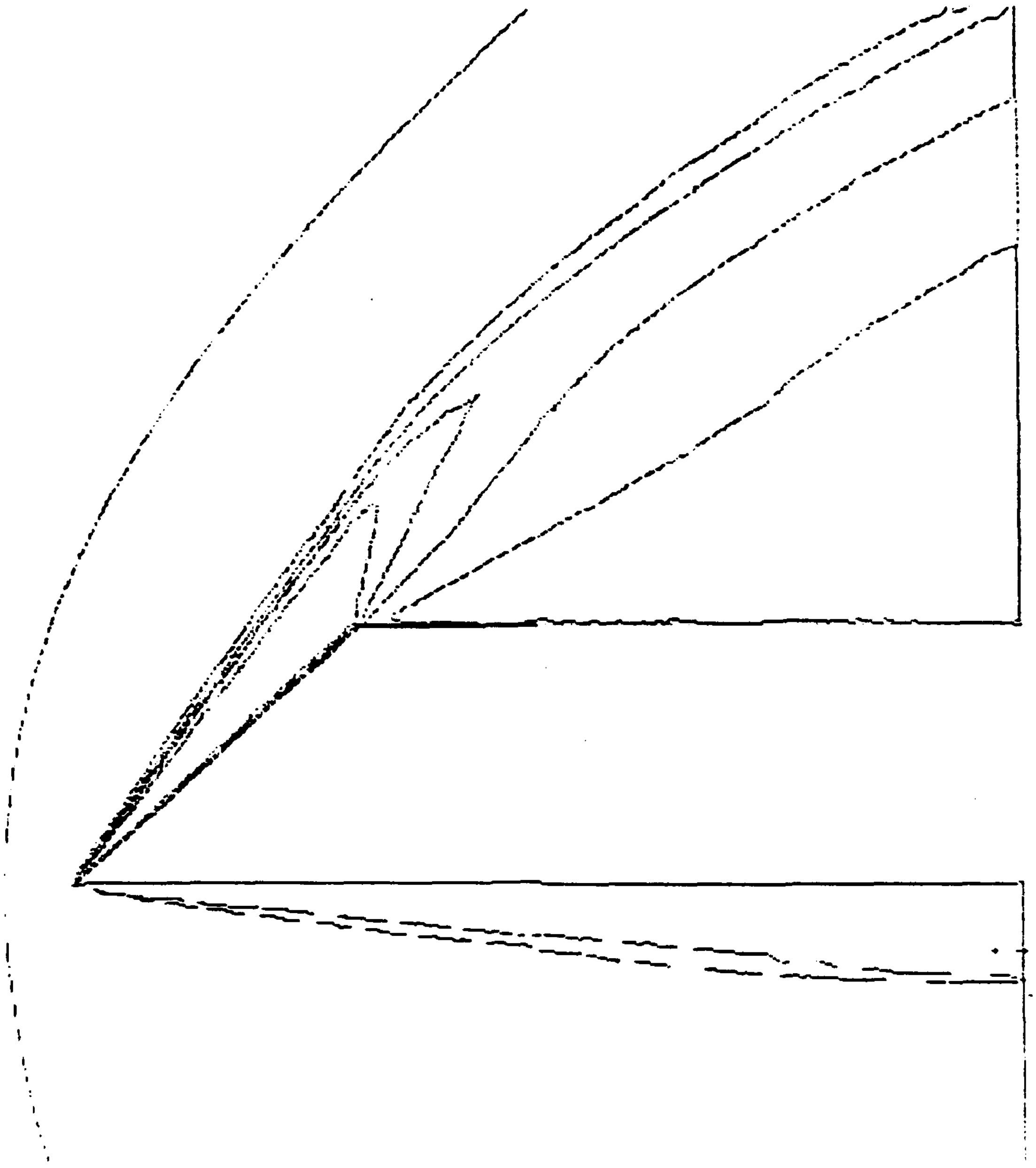


Figure 9b. Demonstration Case 1 Low Stagnation Temperature Case - Density Profiles for Comparison With Schlieren (fig. 9a).

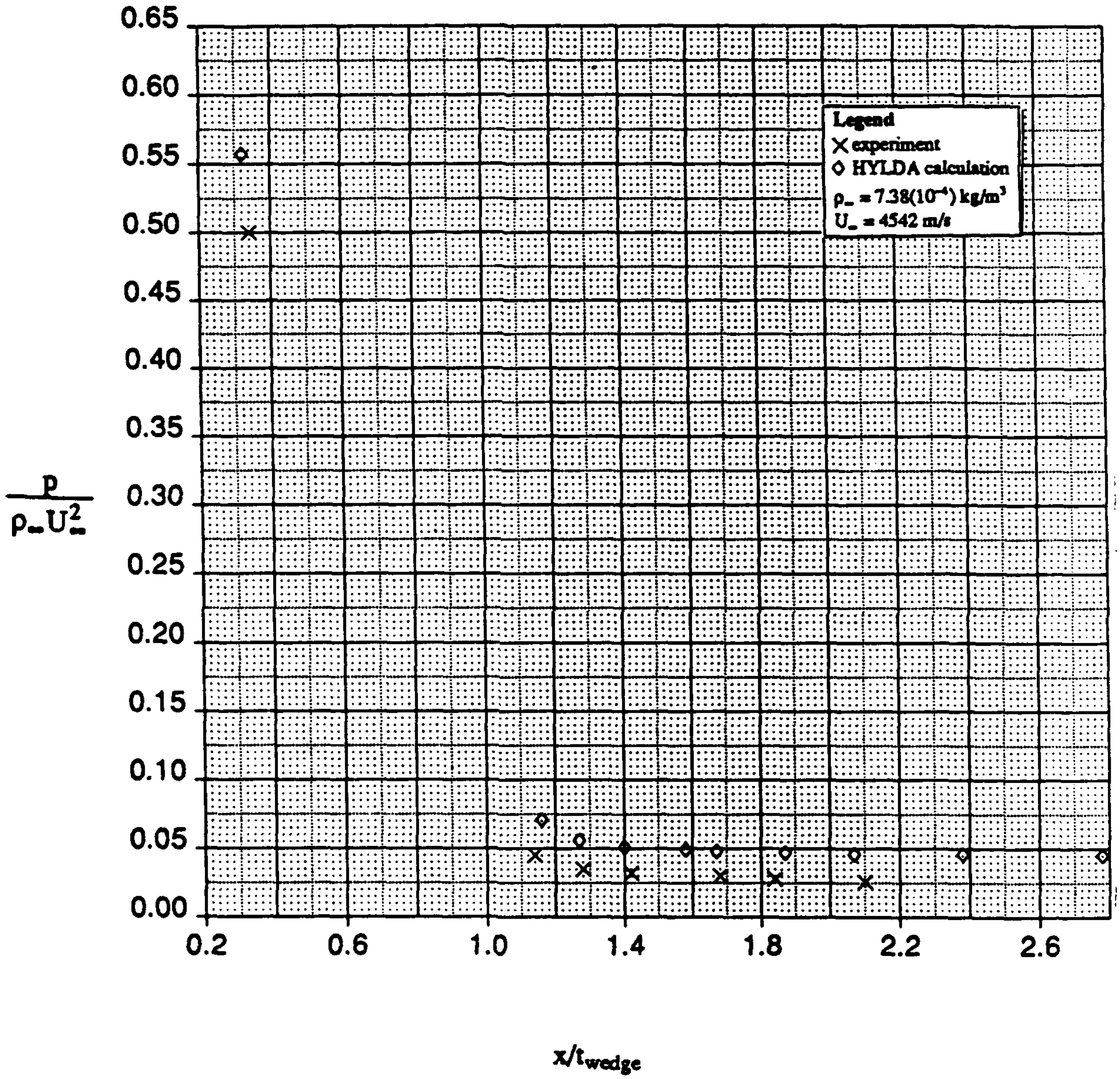


Figure 10. Demonstration Case 1 Low Stagnation Temperature Case - Pressure

High Stagnation Temperature Results

The grid and the computational domain for the high stagnation temperature case, shown in fig. 11, are similar to those used for the low stagnation temperature case. Using the shock structure from the low stagnation temperature case as an example, the grid for the high stagnation temperature case was reduced from the 80x50x1 used for the low temperature case to a 70x35x1 grid, with grid points removed primarily from the freestream section of the original 80x50x1 grid. Again, only one cell is required in the third direction since the flow is essentially two-dimensional. The grid is still clustered near the nose of the wedge to resolve the shock, and also clustered near the body to resolve the viscous layers.

The high stagnation temperature calculation was accomplished in 448 steps with a maximum CFL number of 1000. Figs. 12 through 14 show the Mach, temperature, and γ contours for this case, where γ is the ratio of specific heats. In comparing to the low stagnation temperature case discussed above (figs. 6 through 8), we see that the features of the flowfield are qualitatively similar. The shock is closer to the body in the high stagnation temperature because the Mach number is slightly higher. The maximum temperature for this case is near 5200K compared to 2300K for the low stagnation temperature case.

The γ contours provide an indication of the nonequilibrium effects of this higher temperature, higher Mach number test case. In comparing to the corresponding low-temperature γ contours, we see that the nonequilibrium flow in the low stagnation temperature case leads to a γ of 1.289, while the higher temperature case leads to a γ of 1.270, showing that the nonequilibrium effects are slightly larger in the high stagnation temperature case. In both cases, the maximum nonequilibrium effects occur behind the shock along the wedge face.

As noted earlier, direct comparisons are not strictly applicable here because of the difference in initial conditions between the experiment and the calculation. However, the difference in conditions is not large enough to completely preclude at least a qualitative comparison of the surface data obtained from these tests. Figs. 15 and 16 show a comparison of the experimental surface pressure and heat flux measurements with the HYLDA results for those same quantities. Also included in the figures are the results of an ideal gas calculation of the high-temperature case using HYLDA.

Fig. 15 shows the surface pressures from the experimental study as well as the numerical results from the ideal gas and finite rate calculations. The figure shows that the largest discrepancy between the experimental results and the calculated results occurs at the single data point on the wedge face (x/t_{wedge} ~ 0.4), with the ideal gas value falling between the experimental data point and the finite rate chemistry calculation. Less discrepancy is found along the flat plate afterbody, but again the ideal gas calculation falls between the experimental and finite rate chemistry calculation results.

Fig. 16 shows the surface heat flux along the body, again for the experimental results and both the ideal gas and finite rate chemistry calculations. This figure shows greater sensitivity to the difference in freestream conditions, with three distinct and widely varying heat flux results. The ideal gas results show the greatest surface heat flux, as expected. The finite rate chemistry calculation shows the correct trend of decreasing heat flux since the nonequilibrium chemistry effects extract some of the energy from the flow. The experimental investigation, however, shows a significantly lower degree of heat flux.

Possible explanations for the wide differences in heat fluxes shown in fig. 16 include the differences in the freestream velocity field, the transient response of the heat flux gauges used in the experiment, and lack of resolution of the thermal boundary layer in the numerical calculations. Additional study is required to determine the exact nature of the discrepancy.

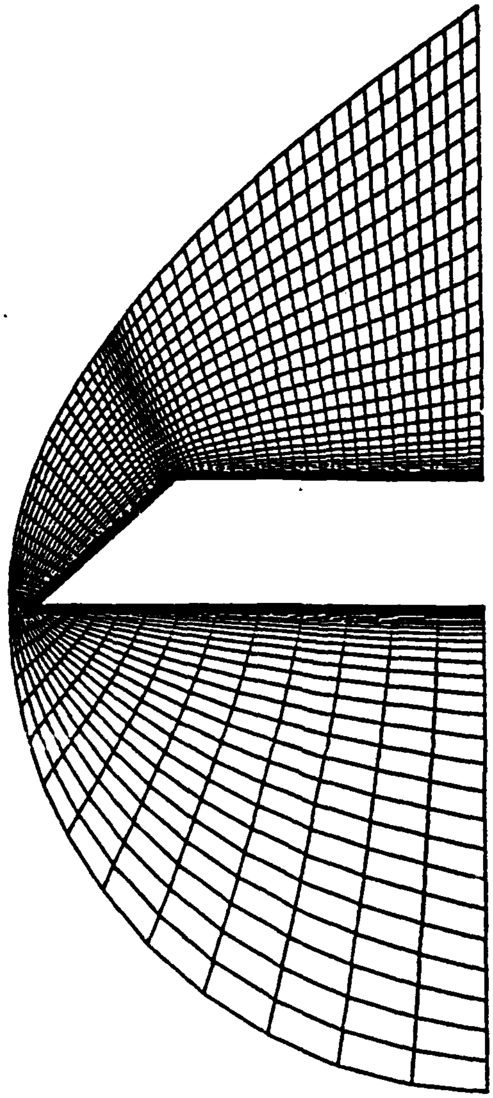
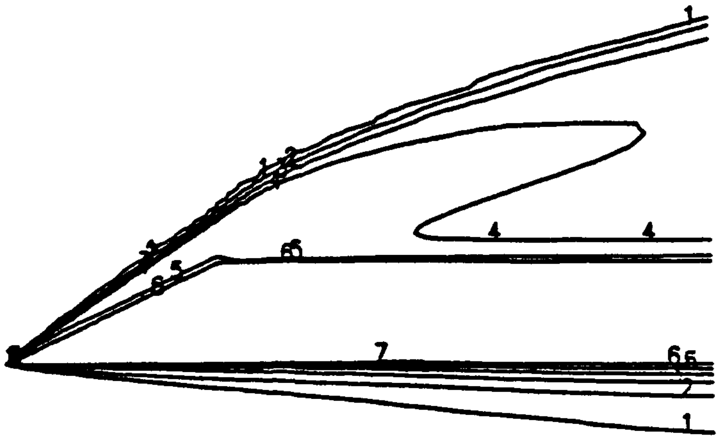
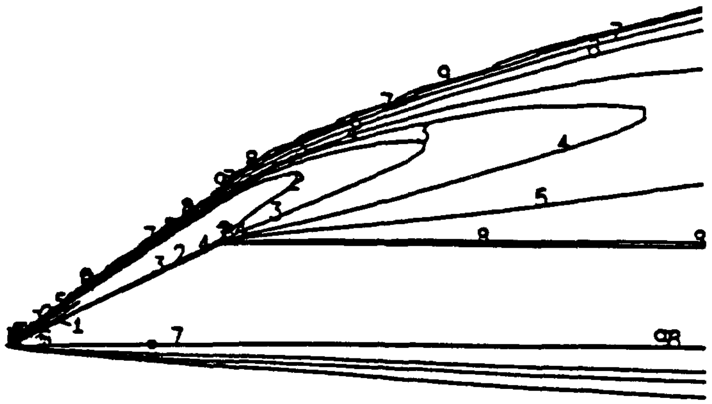


Figure 11. Demonstration Case 1 Mesh for High Stagnation Temperature Case



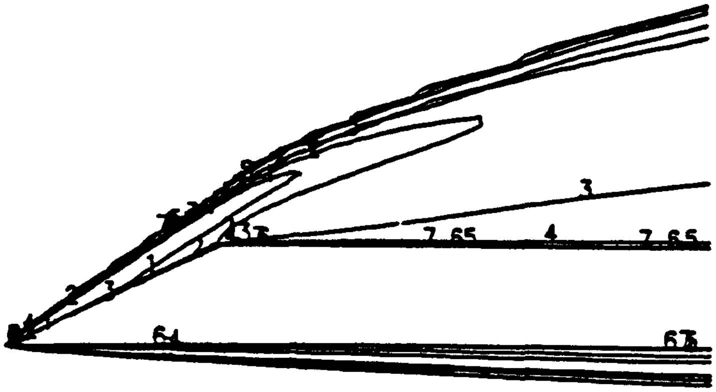
Legend - Mach Number	
1	16.3
2	10.0
3	5.0
4	3.0
5	2.0
6	1.0
7	0.0

Figure 12. Demonstration Case 1 High Stagnation Temperature Case - Mach Contours



Legend - T / T_∞	
1	61.17
2	53.24
3	47.32
4	41.17
5	35.29
6	23.54
7	11.77
8	5.88
9	3.54
T_∞	85 K

Figure 13. Demonstration Case 1 High Stagnation Temperature Case - Temperature Contours



Legend - γ	
1	1.276
2	1.280
3	1.290
4	1.300
5	1.350
6	1.375
7	1.390

Figure 14. Demonstration Case 1 High Stagnation Temperature Case - γ Contours

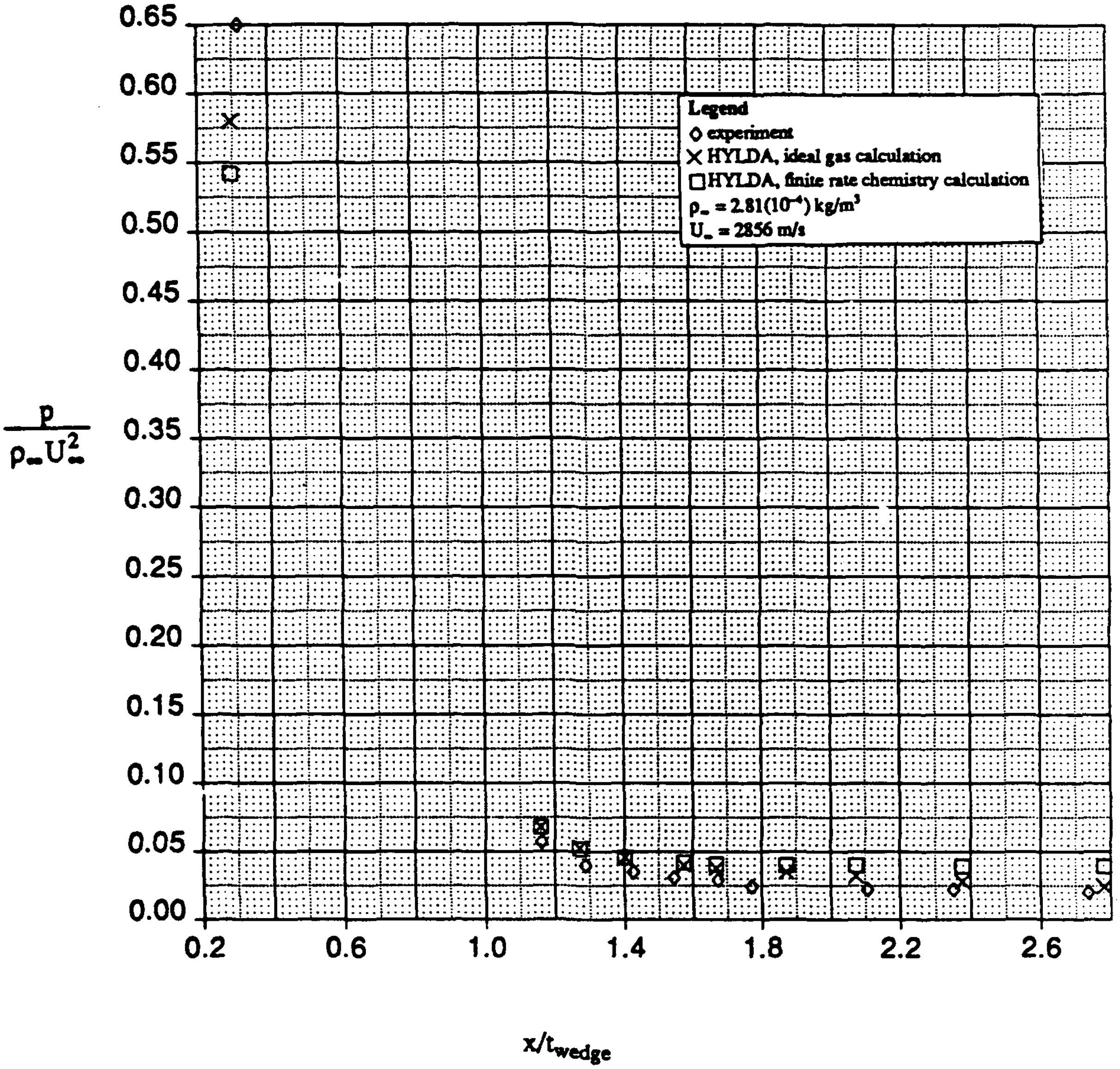


Figure 15. Demonstration Case 1 High Stagnation Temperature Case - Pressure

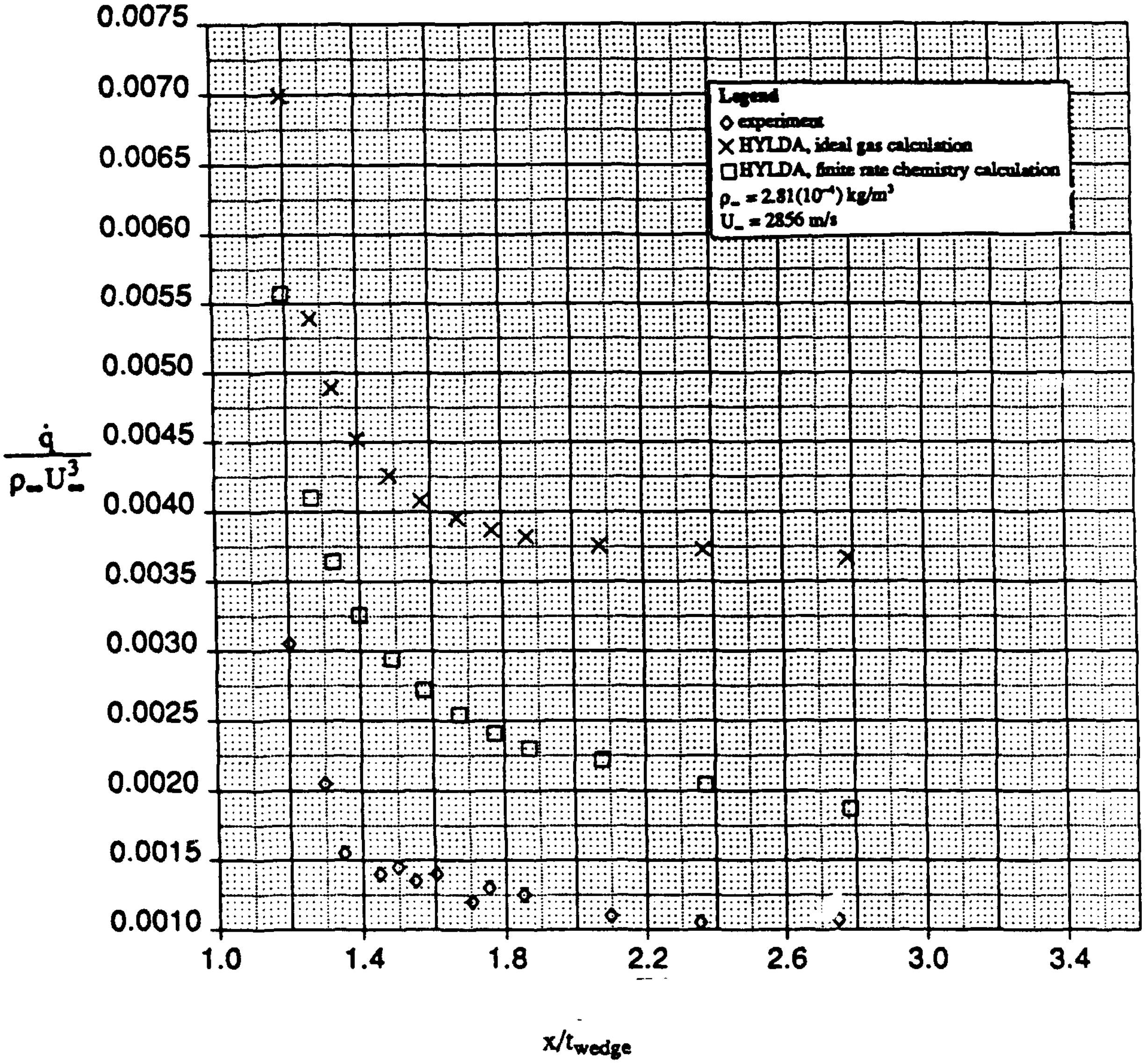


Figure 16. Demonstration Case 1 High Stagnation Temperature Case - Heat Flux

5.2 Case 2: Mach 6.9 Biconic Flow

Validation case 2 investigated the surface heat flux along a straight biconic at Mach 6.9. The biconic geometry is depicted in fig. 17 and was oriented at an angle of attack of zero degrees to the flow. The fluid medium was air and the experiment was performed at NASA Langley's Expansion Tube Facility (ref. 34). The freestream velocity, pressure, and temperature are shown in Table 6 below. Nonequilibrium effects near the stagnation region, if any, are expected to be small and a seven-species finite-rate air chemistry model (ref. 35) was employed to evaluate HYLDA's nonequilibrium heat transfer capabilities.

Table 6. Demonstration Case 2 Initial Conditions

U_∞	P_∞	T_∞	Fluid
5326 m/s	2182 N/m^2	1604K	Air

The computational grid for this test case is shown in fig. 18. The grid included 219 cells along the wetted length, 30 cells normal to the body and 1 cell in the circumferential direction. A high grid density was employed along the body and near the wall since the primary interest of this case was to accurately predict the surface heat flux. Numerical simulation of the expansion tube test was performed with the measured wall temperature distribution shown in fig. 19. This distribution was input into the HYLDA code by modifying the face 5 specified temperature wall boundary condition through the use of a data statement. This modification was specific to this particular demonstration case and was not delivered with the HYLDA code.

Fig. 20 shows a comparison of the predicted wall heat flux with measured values at the 0- and 180-degree meridians. The predictions are generally lower than measurements along both conic sections. A parabolized Navier-Stokes prediction (ref. 34) yielded slightly improved agreement with measurements along the first cone; however, very similar results were obtained along the second conic section. Ref. 34 suggests several experimental uncertainties that also may have influenced the experimental accuracy such as uncertainties in the freestream conditions and the establishment of a non-uniform flowfield. Computational uncertainties regarding grid density effects must also be considered. Additional study is required to determine the exact nature of the anomalies in the predictions.

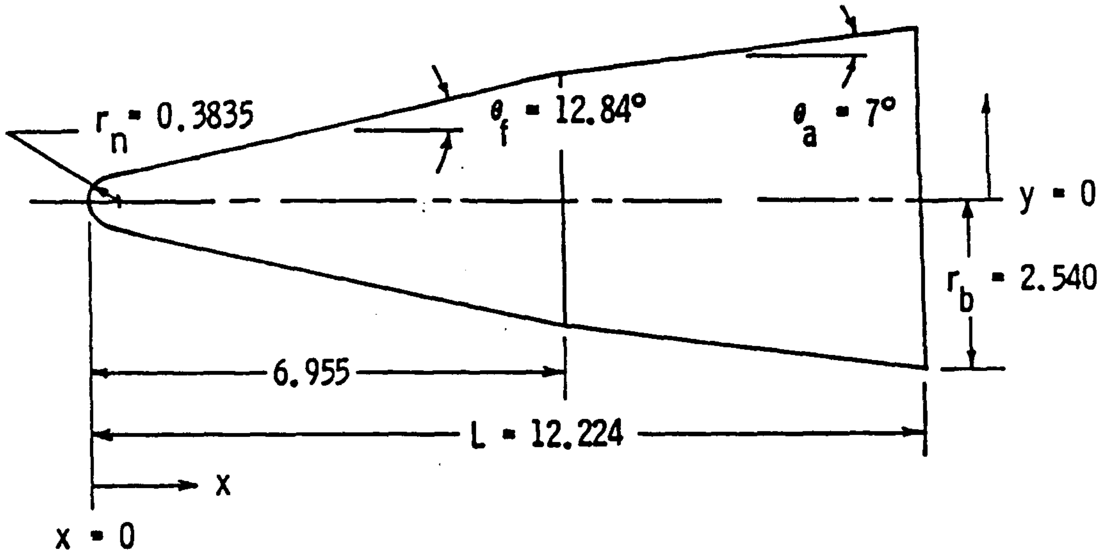


Figure 17. Demonstration Case 2 Geometry

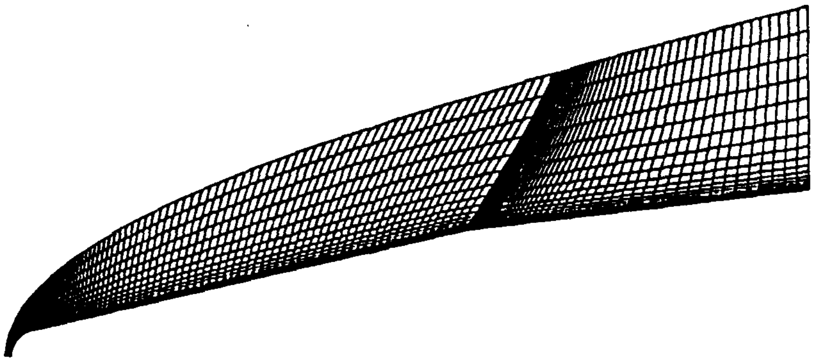


Figure 18. Demonstration Case 2 Mesh

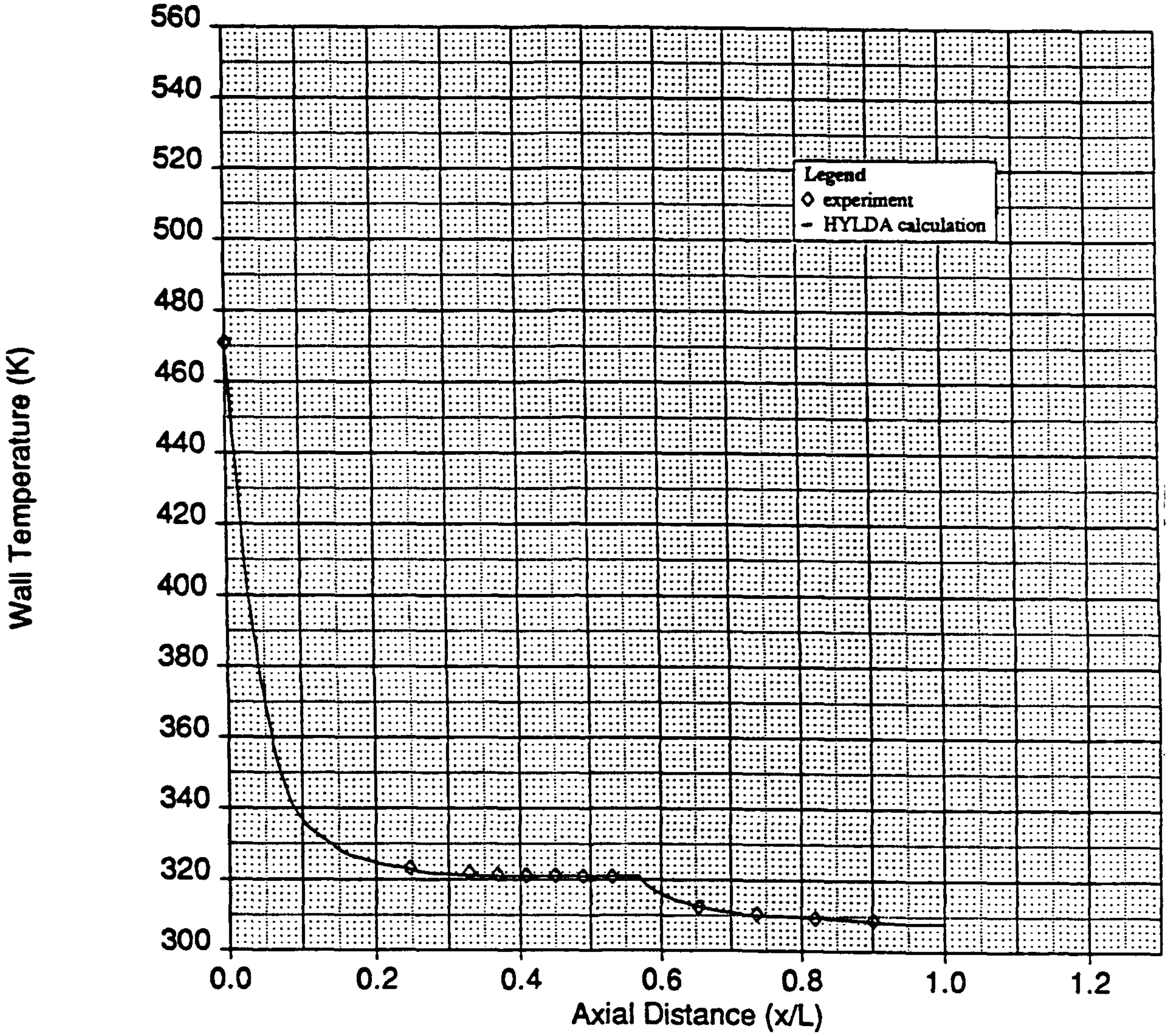


Figure 19. Wall Temperature Along the 180-Degree Meridian of the Straight Biconic

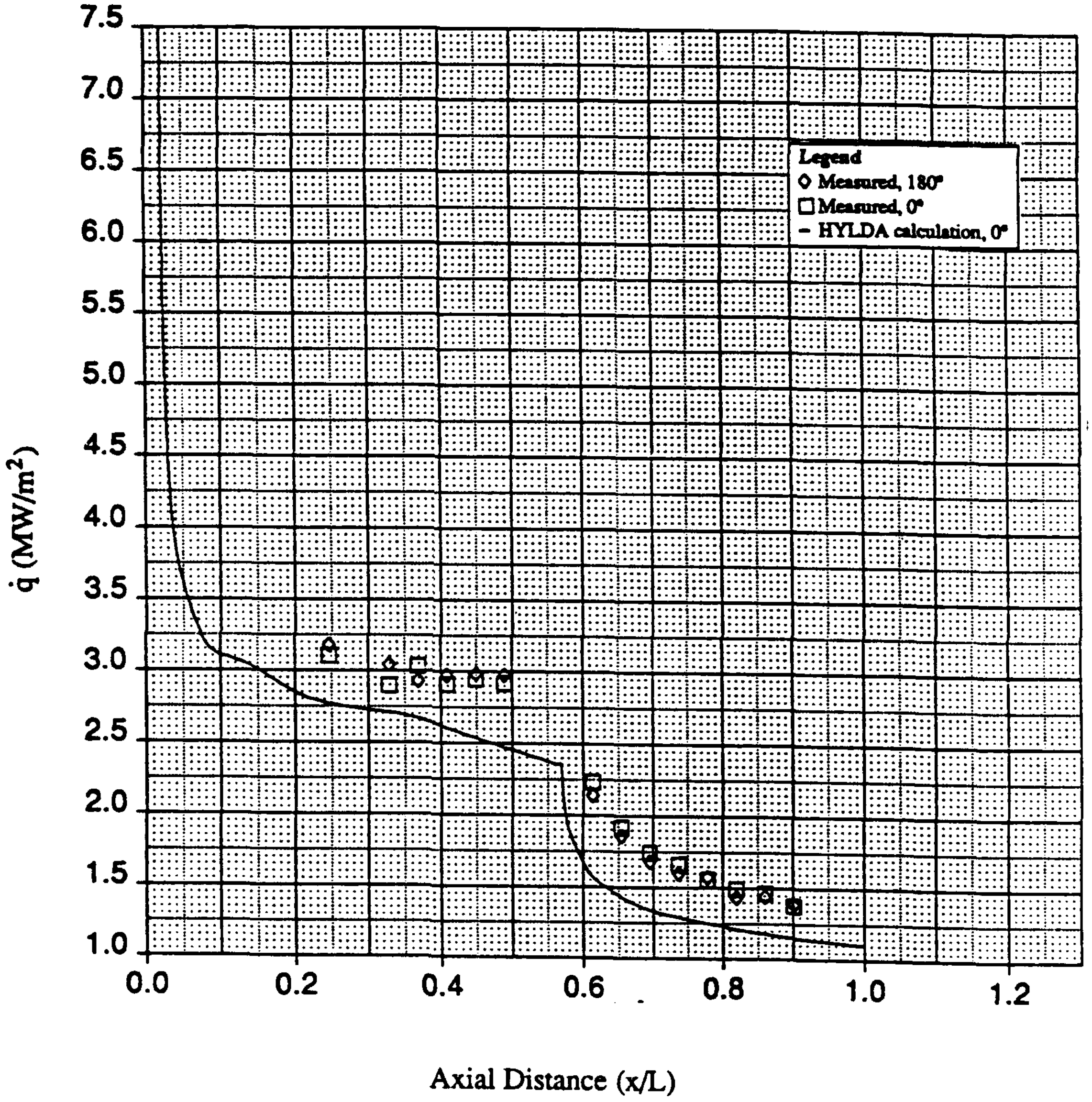


Figure 20. Heat Flux Comparison for the Straight Biconic

5.3 Case 3: Mach 5 Catalytic Blunt Sphere Flow

The third demonstration case was based on an arc jet study of a catalytic hemisphere in a Mach 5 reacting Nitrogen flow (ref. 36). The purpose of this case was to compare the stagnation point heat flux calculated by HYLDA with the experimentally-obtained value. This single point of comparison proved difficult to match numerically, so the study of this case was expanded beyond the single data point comparison to include:

- Comparison of stagnation point heat flux,
- Effects of wall catalycity on stagnation point heat flux,
- Effects of grid clustering on stagnation point heat flux, and
- Mixture effects on stagnation point heat flux.

This expanded approach also provides insight into the code sensitivities to both grid and physical effects.

Geometry and Initial Conditions. The geometry used for this case was a 5.08-cm (2.0-inch) nose radius hemisphere. The initial conditions provided by tables 1 and 2 in ref. 36, supplemented by additional parameters provided upon request by the ref. 36 authors were as follows:

Table 7. Demonstration Case 3 Initial Conditions

$M_\infty = 5.4$
$U_\infty = 5797$m/sec (19020 ft/sec)
$P_\infty = 1.31(10^{-4})$ atm
$T_\infty = 2920$ K (5256 R)
$X_{N_2} = 0.6$
$X_N = 0.4$

As was the case for demonstration cases 1 and 2, it was difficult to determine the required HYLDA input parameters from the discussion of the experiment in ref. 36; thus, there is some uncertainty as to the accuracy with which the numerical calculation models the actual experiment. This difficulty to ensure that the numerical calculation accurately represents the experimental conditions is a problem that should be addressed when performing any new hypersonic wind tunnel tests so as to provide adequate information about the test to allow accurate numerical modeling.

The grid consisted of a 54x30x1 mesh, with 54 cells in the streamwise direction, 30 cells in the radial direction, and 1 cell in the circumferential direction since the flow was essentially axisymmetric. The grid is shown in fig. 21 below:

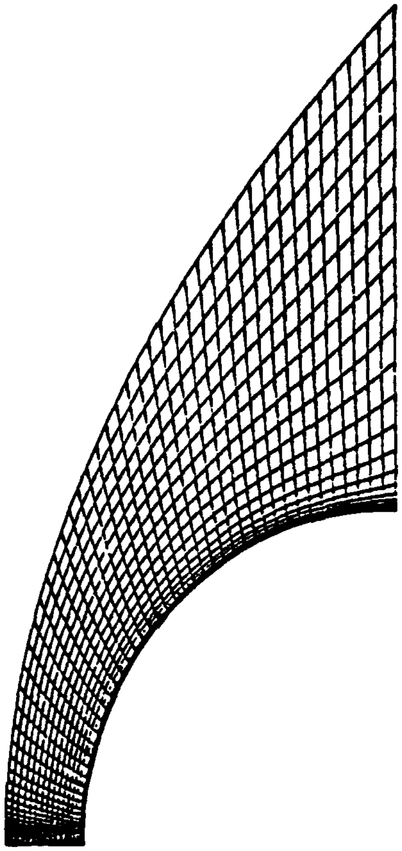


Figure 21. Demonstration Case 3 Mesh

Results. As noted above, the actual comparison to experimental data was limited to a single data point. The poor comparison with the experimental results resulted in an expansion of the case to include studies on the effects of grid and physical parameters on the heat flux results. The four parts to the expanded study are discussed separately below.

Comparison of Stagnation Point Heat Flux. The first attempt to compare with the experimental data consisted of a single data point for the stagnation point heat flux on the hemisphere, using a wall catalytic efficiency of 0.01. The experimental data point was obtained from fig. 2 of ref. 36, and the numerical data point was obtained using HYLDA. For the specified catalytic efficiency of 0.01, the experimental value of the stagnation point heat flux was 0.49(10^6) W/m^2, and the calculated value was 1.77(10^6).

There is a large difference between the experimental and numerical values, and it is not known if the discrepancy is due to an inaccurate modeling of the flow conditions, or an error in the calculation. To study the possibilities for error, additional data points were considered. In this case, ref. 36 provided some indication of the stagnation point heat transfer expected for various other catalytic efficiencies. These data points were considered next.

The same geometry and grid were used to calculate the stagnation point heat flux for catalytic efficiencies of γ_s = 0.00, 0.01, 0.50, and 1.00. The results of these calculations are shown in fig. 22 below, with corresponding experimental data points obtained from fig. 2 in ref. 36.

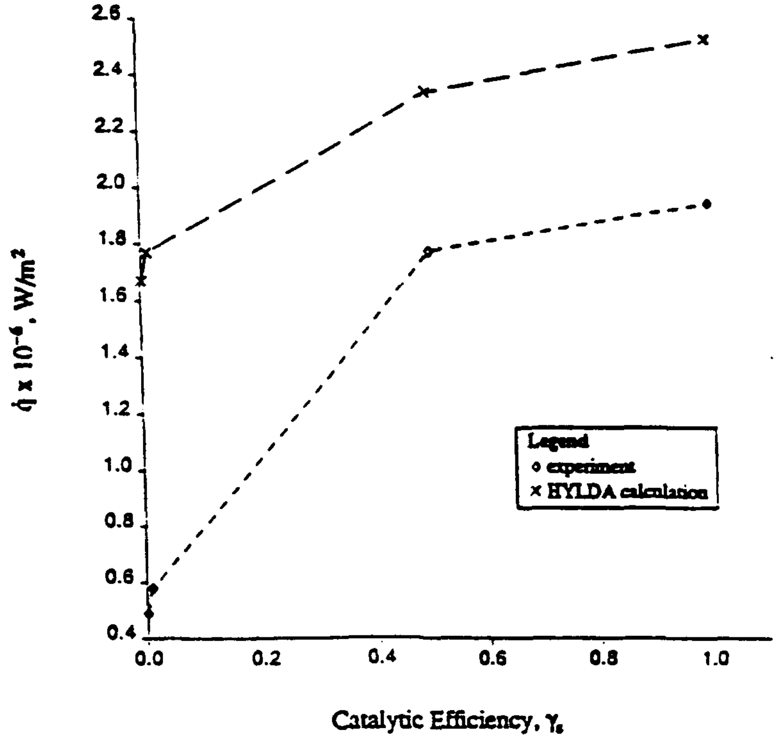
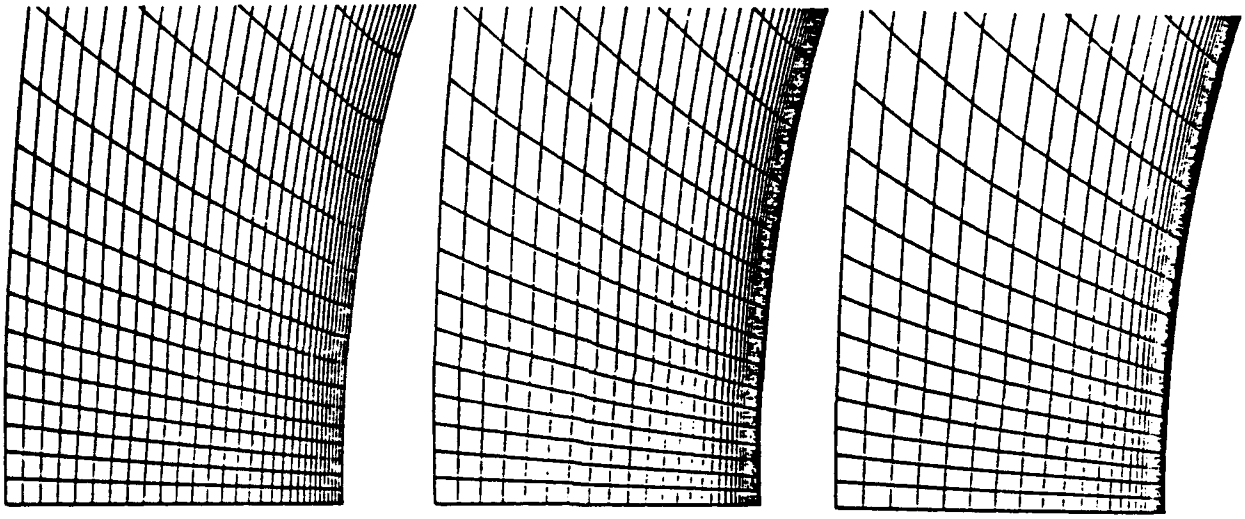


Figure 22. Stagnation Point Heat Flux for a Range of Catalytic Efficiencies

Fig. 22 shows that the HYLDA-predicted stagnation heat flux compares more closely to the experimental values at the higher-catalyticity values, while overpredicting considerably the low-catalyticity cases. The dotted lines in the figure were added only to distinguish the data points. Because the noncatalytic case is the most basic case, and because the largest discrepancies occurred for the noncatalytic case, the noncatalytic case was chosen for further study.

Effects of Grid Clustering on Heat Flux. The third test consisted of calculating the stagnation point heat flux for the noncatalytic case using grids with varying clustering of grid points near the wall in the stagnation region. There is strong evidence to suggest that grid clustering near the wall can significantly effect the calculated heat flux (ref. 37). Fig. 23 (a-c) shows the nose region grids for the three different grid clusterings used for this test, ranging from coarse (grid a) to fine (grid c).



a. Coarse　　b. Medium　　c. Fine

Figure 23. Grid Clustering Schemes

Table 8 shows the results of the analysis, including the grid spacing of the first cell at the stagnation point and the associated calculated heat flux at the stagnation point. Also included in the table is the noncatalytic value from fig. 2 in ref. 36.

Table 8. Grid Clustering Effects on Stagnation Point Heat Transfer

Grid	First Cell Height (m)	Stagnation Heat Flux (W/m^2)
a	$6.21(10^{-5})$	$1.37(10^{6})$
b	$2.71(10^{-5})$	$1.44(10^{6})$
c	$1.43(10^{-5})$	$1.61(10^{6})$
Experiment	-	$0.50(10^{6})$

Table 8 shows that grid spacing near the wall does have some effect on the stagnation point heat flux, but in this case it is not significant enough to be considered the only cause of the discrepancy between the calculated and experimental results.

In a final attempt to quantify the difference between experimental and numerical results, the effect of the freestream mixture properties was investigated.

Mixture Effects on Heat Flux. Since it is often difficult to determine accurately the freestream conditions in hypersonic wind tunnel tests, one possible source of inaccuracy could be related to the presence of a different freestream than that reported for the wind tunnel test. This final calculation was directed at studying the effect that a different freestream mixture would have on the stagnation point heat flux. This calculation used the same 54x30x1 grid shown in fig. 21, and the same conditions shown in table 7, with the exception that the freestream was assumed to consist of pure N_2 rather than the mixture of N_2 and N as reported in ref. 36. The results for this calculation are plotted in fig. 24:

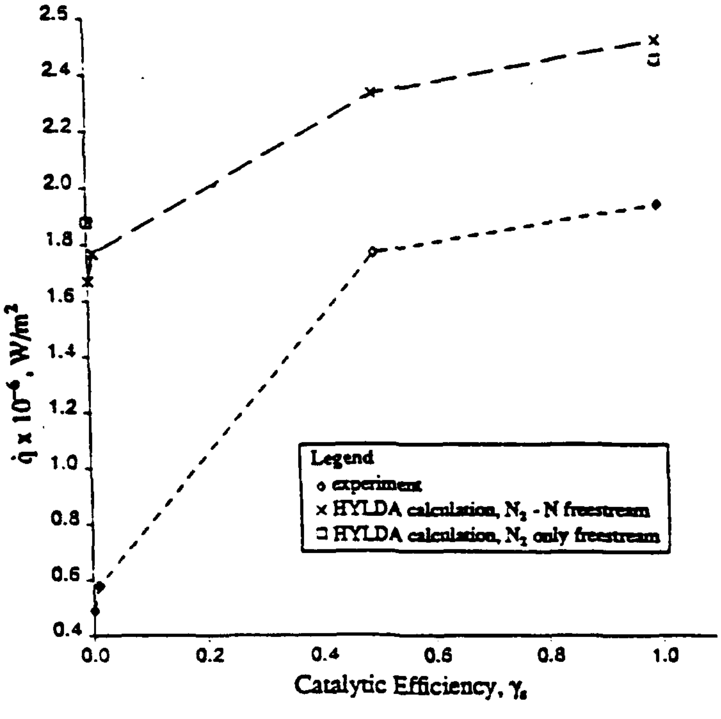


Figure 24. Freestream Effects on Stagnation Point Heat Flux

Fig. 24 is the same as fig. 22, with the new data points for the pure N_2 freestream added. Again, dotted lines were added to the data points from fig. 22 to distinguish data points. Dotted lines were not drawn for the new data points, since only two catalytic efficiencies (0.00 - noncatalytic, and 1.00 - fully-catalytic) were calculated. Fig. 24 shows that the freestream effects for this calculation were not significant at the fully-catalytic value, decreasing slightly the previous HYLDA-calculated value using the ref. 36 reported freestream. The new value for the noncatalytic case increased over the previous HYLDA calculation, leading to worse comparison with the experimental value. In general, though, the effects of the different freestream were not large, and cannot be interpreted as the cause of the discrepancy between the numerical and experimental results.

The goal of this demonstration case was initially one of comparing stagnation point heat flux for a catalytic blunt sphere. However, the discrepancy between the numerical and experimental results led to a study of various effects on heat flux, including catalyticity effects, grid effects, and freestream mixture properties effects. While none of these effects is the cause of the discrepancy, each of these studies did effect the solution to some degree. Further, the uncertainty in the exact degree to which the numerical calculation accurately represented the experimental test is not clear, and precludes additional testing on this particular case.

6.0 CONCLUSIONS AND RECOMMENDATIONS

The HYLDA code developed for this contract is capable of predicting trends in hypersonic flowfields including real gas effects on heating rates, pressures, forces, and moments on complex shapes in low-density flows. The code is based on a fully-coupled Navier-Stokes/finite rate chemistry algorithm, using an implicit solution algorithm and a one temperature chemistry model. The species thermodynamics properties model, and the transport properties model have been tailored to fit the requirements for calculating finite rate air chemistry up to Mach 25.

The HYLDA code is a versatile code capable of calculating ideal gas flows, equilibrium air flows, and finite rate reacting air flows. The code is written in a top-down format with a flexible data structure, both of which facilitate the incorporation of new technology into the code as it becomes available. The chemistry model, species thermodynamics model, and the transport properties model are also flexible, allowing different chemistry models to be installed with minimal changes.

The demonstration cases included in this report are not an attempt to fully validate the HYLDA code. They do, however, demonstrate some of the capabilities of the code to calculate finite rate chemistry flows, while at the same time accentuating the need for new wind tunnel data on hypersonic flows to support the validation efforts of new Navier-Stokes/finite rate chemistry codes. The demonstration cases provide insight into the approach used to calculate hypersonic reacting flows, and provide details of the difficulty in trying to numerically reproduce experimental results. Full validation of the HYLDA code will require calculation of a large number of cases over a wide range of conditions, and will require a significant amount of time to complete. The Air Force has already begun efforts to validate the HYLDA code.

Recommendations for future work with the HYLDA code can be grouped into four categories:

- Validation studies,
- Boundary condition extensions,
- Algorithm extensions, and
- Alternate and improved chemistry models.

Validation studies should include both numerical-to-experimental comparisons and numerical-to-numerical comparisons. Meaningful numerical-to-experimental comparisons will require careful analysis of existing experimental hypersonic analyses, and greater emphasis on new and improved methods for obtaining accurate hypersonic experimental data using techniques such as those described in ref. 1. Numerical-to-numerical comparisons will provide a measure of the effects of different numerical techniques, in particular the chemistry models being used. As new models become available, codes such as HYLDA that are able to rapidly and efficiently incorporate new models will facilitate the analysis of the models.

Boundary condition extensions for the HYLDA code should include the addition of multizone capability to allow calculation of more complex geometries than those being considered in the current contract effort. Extensions to the physical boundary condition models should include porous walls (to include transpiration and/or suction) and improved catalytic models. The flexible data structure of the HYLDA code will enable incorporation of these and other boundary condition improvements without requiring major code revisions.

Algorithm improvements to HYLDA should include incorporation of new developments intended to make the algorithm more stable (to save computer time) and to increase code robustness (to extend the range of applicability). These improvements should consider alternative methods of flux splitting, improved· use of multiprocessing capabilities, and adaptive-chemistry capability. Again, the HYLDA data structure should facilitate the incorporation of new technology.

Chemistry models are the most dynamic models in use in the HYLDA code. Many special-purpose chemistry models exist, and could be incorporated into HYLDA. Many other air chemistry models exist, and these models could be incorporated into HYLDA to facilitate direct comparisons between HYLDA and other finite rate air chemistry codes. Of particular interest should be additional transport properties models and species thermodynamics databases.

The flexibility of the HYLDA Navier-Stokes/finite rate chemistry code should make it a valuable design tool for the next-generation of hypersonic vehicles being developed.

7.0 REFERENCES

1. Cassady, P.E., Lieberg, S.F., "Planar Laser-Induced Fluorescence Measurements in Hypersonic Shock Tunnel Flow," AIAA 90-1549, June 1990.

2. Hoffman, J.J., Wong, R.S., Bussing, T.R., and Birch, S.F., "Low Density Real Gas Flows About Hypersonic Vehicles: Interim Report," AFWL-TR-87-3112, March 1988.

3. "Software User's Manual for the Hypersonic Low Density Analysis (HYLDA) Code," WL-TR-91-3060, AFWL Contract F33615-86-C-3006, December 1990.

4. Baldwin, B.S., and Lomax, H., "Thin Layer Approximation and Algebraic Model for Separated Turbulent Flows," AIAA paper 78-0257, January 1978.

5. Tannehill, J.C., and Mugge, P.H., "Improved Curve Fits for the Thermodynamic Properties of Equilibrium Air Suitable for Numerical Computation using Time-Dependent Shock-Capturing Methods," NASA CR-2470, October 1974.

6. Park, C., and Menees, G.P., "Odd Nitrogen Production by Meteoroids," Journal of Geophysical Research, Vol. 83, No. C8, 1978.

7. Nicolet, et. al., "Analytical and Design Study for a High Pressure, High Enthalpy Constricted Arc Heater," Aerotherm/Acurex Report 74-125, 1975.

8. McBride, B. J., Personal Communication.

9. MacCormack, R.W., "Current Status of Numerical Solutions of the Navier-Stokes Equations," AIAA paper 85-0032, AIAA 23rd Aerospace Sciences Meeting, January 1985.

10. Steger, J.L., and Warming, R.F., "Flux Vector Splitting of the Inviscid Gasdynamic Equations with Application to Finite-Difference Methods," Journal of Computational Physics, Vol. 40, pp 263-293, 1981.

11. Hoffman, J.J., "Development of an Algorithm for the Three-Dimensional Fully-Coupled Navier-Stokes Equations With Finite Rate Chemistry," AIAA-89-0670, AIAA 26th Aerospace Sciences Meeting, January 1989.

12. Hopcroft, R.G., "Structuring Computational Fluid Dynamics Analysis Systems," presented at the Third International Conference on Numerical Methods in Laminar and Turbulent Flows, August, 1983.

13. Krauss, J.K., and Patterson, J.C., "Case Study: The Parallelization of HYLDA," Boeing Computer Services Engineering Computing and Analysis Technical Report ECA-TR-151, December 1990.

14. Candler, G.V., "The Computation of Hypersonic Ionized Flows in Chemical and Thermal Nonequilibrium," AIAA paper 88-0511, 26th Aerospace Sciences Meeting, January 1988.

15. Wong, R.S., and Bussing, T.R., "A Simplified, One-Temperature Air Chemistry-Kinetics Model," AIAA paper 89-0198, 27th Aerospace Sciences Meeting, January 1989.

16. Kee, R.J., and Dwyer, H.A., "Review of Stiffness and Implicit Finite-Difference Methods in Combustion Modeling," presented at the 7th ICOGER, Gottingen, Federal Republic of Germany, 1979.

17. Beam, R.W., and Warming, R.F., "An Implicit Finite-Difference Algorithm for Hyperbolic Systems in Conservation-Law Form," Journal of Computational Physics, Vol. 22, pp 87-110, 1976.

18. Jameson, A., Schmidt, W., and Turkel, E., "Numerical Solutions of the Euler Equations by Finite Volume Methods Using Runge-Kutta Time-Stepping Schemes," AIAA 81-1259, AIAA 14th Fluid and Plasma Dynamics Conference, 198..

19. Yee, H.C., "Upwind and Symmetric Shock-Capturing Schemes," NASA Technical Memorandum 89464, May 1987.

20. MacCormack, R.W. and Candler, G.V., "The Solution of the Navier-Stokes Equations Using Gauss-Seidel Line Relaxation," presented at the Symposium in Honor of G. Moretti's 70th Birthday," Polytechnic University, Long Island, 1987.

21. Grossman, B., and Cinnella, P., "The Development of Flux-Split Algorithms for Flows With Non-Equilibrium Thermodynamics and Chemical Reactions," AIAA paper 88-3595-CP, 1988.

22. Degani, D., and Shiff, L.B., "Computation of Supersonic Viscous Flows Around Pointed Bodies at Large Incidence," AIAA 83-0034, January 1983.

23. Camac, M., and Feinberg, R.M., "Formation of NO in Shock-Heated Air," 11th Symposium (International) on Combustion, 1966.

24. Lin, S., Neal, R., and Fyfe, W., "Rate of Ionization Behind Shock Waves in Air: Part I. Experimental Results," Physics of Fluids, Vol. 5., No. 12, 1962.

25. Wray, K., "Chemical Kinetics of High Temperature Air," Hypersonic Flow Research, 1962.

26. Kang, S.W., and Dunn, M.G., "Theoretical and Experimental Studies of Reentry Plasmas," NASA CR-2232, 1973.

27. Bittker, D.A., and Scullin, V.J., "GCKP84-General Chemical Kinetics Code for Gas-Phase Flow and Batch Processes Including Heat Transfer Effects," NASA Paper 2320, 1984.

28. Bortner, M.H., "A Review of Rate Constants of Selected Reactions of Interest in Reentry Flow Fields in the Atmosphere," National Bureau of Standards Technical Note 484, 1969.

29. Brokaw, R.S., Journal of Chemical Physics, Vol. 29, p. 391, 1958.

30. Kee, R.J., Warnatz, J., and Miller, J.A., "A Fortran Computer Code Package for the Evaluation of Gas-Phase Viscosities, Conductivities, and Diffusion Coefficients," Sandia Lab Report SAND 83-8209, 1983.

31. Curtiss, C.F., Hirschfelder, J.O., "Transport Properties of Multicomponent Gas Mixtures," Journal of Chemical Physics, Vol. 12, No. 6, 1949.

32. Mrozinski, D.P., and Hoffman, J.J., "A Full Navier-Stokes Code for Hypersonic Nonequilibrium Flowfield Calculations," presented at the 4th International Symposium on Computational Fluid Dynamics, 9-12 September 1991 at Davis, California.

33. Vidal, R.J., and Stoddard, F., "Measurements of Nonequilibrium Effects in Air on Wedge-Flat-Plate Afterbody Pressures," NASA CR-328, November 1965.

34. Miller III, C.G., Micol, J.R. and Gnoffo, P.A., "Laminar Heat Transfer Distributions on Biconics at Incidence in Hypersonic-Hypervelocity Flows," NASA TP-2213, 1985.

35. Vincenti, W.G. and Kruger, C.H., "Introduction to Physical Gas Dynamics," John Wiley and Sons, 1965.

36. Kolodziej, P., and Stewart, D.A., "Nitrogen Recombination on High-Temperature Reusable Surface Insulation and the Analysis of Its Effect on Surface Catalysis," AIAA-87-1637, AIAA 22nd Thermophysics Conference, June 1987.

37. Hoffmann, K.A., Siddiqui, M.S., and Chiang, S.T., "Difficulties Associated With the Heat Flux Computations of High Speed Flows by the Navier-Stokes Equations," AIAA 91-0467, 29th Aerospace Sciences Meeting, January 1991.

ABBREVIATIONS

AFWL	Air Force Wright Laboratories
CPU	central processing unit
fig	figure
G-S	Gauss-Seidel
HYLDA	Hypersonic Low Density Analysis code
ref	reference
3D	three-dimensional

NOMENCLATURE

English

$\mathbf{a}_1^{n+1}, \mathbf{a}_2^{n+1}, \mathbf{a}_3^{n+1}$	implicit coefficient matrices, eq. (3-18)
$\mathbf{a}_4^{n}, \mathbf{a}_5^{n}, \mathbf{a}_6^{n}$	explicit coefficient matrices, eq. (3-18)
$\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}$	inviscid flux Jacobians, eq. (3-13)
$\mathbf{A}^+, \mathbf{B}^+, \mathbf{C}^+$	positive eigenvalue flux Jacobians, eqs. (3-14), (3-16), (3-17)
$\mathbf{A}^-, \mathbf{B}^-, \mathbf{C}^-$	negative eigenvalue flux Jacobians, eqs. (3-14), (3-16), (3-17)
A^+	constant, eq. (3-38)
C	constant, eq. (3-41)
C_i	constants defined by eq. (4-8), i=1,2
C_{cp}	constant, eq. (3-37)
C_{KLEB}	Klebanoff constant, eq. (3-38)
c_p	specific heat at constant pressure
c_v	specific heat at constant volume
C_{wk}	constant, eq. (3-40)
$D_{s_{mix}}$	species s mixture diffusion coefficient, eq. (3-4)
e	total energy per unit volume = $\rho\left[e_i + \frac{1}{2}\left(u^2 + v^2 + w^2 \right) \right]$, eq. (3-2)

e_i	total internal energy per unit mass, eq. (3-6)
e_{i_s}	species s internal energy, eq. (3-7)
$e_{i_{elec}}$	electronic internal energy, eq. (3-6)
e_i^o	heat of formation, eq. (3-6)
$e_{i_{rot}}$	rotational internal energy, eq. (3-6)
$e_{i_{trans}}$	translational internal energy, eq. (3-6)
$e_{i_{vib}}$	vibrational internal energy, eq. (3-6)
$f_s(T)$	curve-fit function of specific heat at constant pressure
F	Gibbs free energy, eq. (4-3)
$\mathbf{F}', \mathbf{G}', \mathbf{H}'$	x-, y-, z-direction flux vectors, eqs. (3-1), (3-2)
$\mathbf{F}, \mathbf{G}, \mathbf{H}$	ξ–,η–,ζ–direction (transformed) flux vectors, eqs. (3-9), (3-10)
$\mathbf{F}_i, \mathbf{G}_i, \mathbf{H}_i$	inviscid flux vectors, eq. (3-11)
$\mathbf{F}_v, \mathbf{G}_v, \mathbf{H}_v$	viscous flux vectors, eq. (3-11)
$\mathbf{F}_\xi, \mathbf{G}_\eta, \mathbf{H}_\zeta$	transformed viscous flux vectors, eq. (3-17)
F_{KLEB}	Klebanoff intermittency function, eq. (3-38)
F_{VD}	van Driest function, eq. (3-38)
F_{wake}	wake function, eq. (3-39)
F_{w1}	wake function 1, eq. (3-38)

F_{w2}	wake function 2, eq. (3-38)
h	distance normal to the wall, eq. (3-37)
h_f^o	mixture heat of formation
h_s^o	species s heat of formation
h_s	species s enthalpy
H	enthalpy, eq. (4-2)
I	identity matrix, eq. (3-13)
J	Jacobian
k_{boltz}	Boltzmann constant, eq. (4-6)
l_{mj}	direction cosines (m=1-3, j=1-3)
m_i	mass of particle i, eq. (4-6)
M_s	species s molecular weight
M	Mach number
n	time level
n_e	electron number density, eq. (4-10)
$\hat{n}$	normal direction
N	matrix relating nonconservative and conservative solution vectors, eq. (3-13)

N_s	number of species
p,P	pressure
q	<u,v,w>, eq. (3-3)
$\dot{q}$	heat flux
q_e	edge velocity, eq. (3-41)
$\hat{q}_s$	$< \hat{u}_s , \hat{v}_s , \hat{w}_s >$, eq. (3-4)
q_{max}	maximum velocity in the boundary layer, eq. (3-40)
R	rotation matrix, eq. (3-44) mixture gas constant
$\hat{R}$	universal gas constant
R_s	species s gas constant = $\hat{R} / M_s$
Re	Reynolds number
t	time
t_{wedge}	wedge thickness, Demonstration Case 1
T	temperature
(u,v,w)	x-, y-, and z-direction velocity components
$\left[\hat{u}_s , \hat{v}_s , \hat{v}_s \right]$	species diffusion velocities in the x-, y-, and z-directions, eqs. (3-2), (3-4)
u^*	compressibility corrected velocity, eq. (3-41)

u_τ	friction velocity, eq. (3-38)
$\mathbf{U}'$	vector of conserved variables, eq. (3-1)
$\mathbf{U}$	transformed vector of conserved variables, eqs. (3-9), (3-10)
$\mathbf{V}$	nonconservative solution vector, eq. (3-23)
$\mathbf{W}$	Cole's tabulated wake function, eq. (3-41)
$\dot{\mathbf{W}}'$	vector of chemistry source terms, eq. (3-1)
$\dot{\mathbf{W}}$	transformed vector of chemistry source terms, eqs. (3-9), (3-10)
(x,y,z)	physical coordinate system
$\mathbf{X}_s$	species s molar concentration, eq. (3-4)
y^+	law-of-the-wall coordinate, eq. (3-38)
$\mathbf{Z}'$	nontransformed boundary condition vector, eq. (3-23)
$\mathbf{Z}$	transformed boundary condition vector, eqs. (3-26), (3-27)
Z_i	curve-fit coefficients, eqs. (4-1) - (4-3)

Greek

α_{ij}	defined by eq. (4-13)
$\delta_{ij.}$	Kronekar delta, eq. (3-3)
δU^{n+1}	implicit delta, eq. (3-12)
δV	nonconservative flux vector delta
$\Delta_{ij}^{(k)}$	defined by eq. (4-7)
Δt	time step
ΔU^n	explicit delta, eq. (3-12)
(ξ,η,ζ)	transformed coordinate system
γ	ratio of specific heats
γ_s	species s catalytic efficiency, eq. (3-31)
Γ	nontransformed boundary condition vector, eq. (3-23)
κ	von Karman constant, eq. (3-37)
λ	thermal conductivity
μ	viscosity, eq. (3-3)
μ_B	bulk viscosity, eq. (3-3)
$\Omega_{ij}^{(k,k)}$	collision cross-section, eqs. (4-7), (4-9), (4-10)
π	constant, eq. (3-41)

ρ_s	density of species s
ρ	total density = summation of species densities
σ	compressibility correction, eq. (3-41)
τ_{ij}	shear stress tensor, eq. (3-3)
$\dot{\omega}_s$	species s production/depletion rate
ω	vorticity, eq. (3-37)
χ	transformed boundary condition vector, eqs. (3-26), (3-27)

Subscript

adj	adjacent cell
bc	boundary cell
e	edge
∞	freestream
KLEB	Klebanoff
max	maximum
mix	mixture
o	stagnation property
SL	sea level
trans	transport
turb	turbulent
VD	van Driest
wall	wall value

Superscript

′	nontransformed (physical) variables
+	positive fluxes
-	negative fluxes
T	transpose

Units

atm	atmosphere
ft	feet
in	inch
J	Joule
kg	kilogram
K	Kelvin
m	meter
N	Newton
psia	pounds (force) per square inch
R	degrees Rankine
sec	second

APPENDIX: THERMOCHEMISTRY MODEL COEFFICIENTS

This appendix contains the coefficients for the thermochemistry models described in section 4.0. Section A.1 contains the coefficients for the chemistry model, section A.2 contains coefficients for the species thermodynamics properties, and section A.3 contains coefficients for the transport properties equations. For details of the format of these tables, refer to the discussion in the Software User's manual (ref. 3).

A.1 Chemistry Model Coefficients

This section contains the default chemistry model delivered with the HYLDA code. The model is the modified Park and Menees model (modified by R.S. Wong and T.R. Bussing, see the Interim Technical Report, ref. 2), with 11 species (including electrons) and 20 reaction paths. Reaction rate coefficients are of the Arrhenius form.

```
MODIFIED PARK AND MENEES FINITE RATE CHEMISTRY MODEL
20
O2  +  N   =  2 O +  N
0  3
                                                   8.25e+19   -1.00    59500
O2  +  NO  =  2 O +  NO
0  3
                                                   2.75e+19   -1.00    59500
O2  +  N2  =  2 O +  N2
0  3
                                                   2.75e+19   -1.00    59500
O2  +  O2  =  2 O +  O2
0  3
                                                   2.75e+19   -1.00    59500
O2  +  O   =  2 O +  O
0  3
                                                   8.25e+19   -1.00    59500
N2  +  NO  =  2 N +  NO
0  3
                                                   3.70e+21   -1.60   113200.
N2  +  O2  =  2 N +  O2
0  3
                                                   3.70e+21   -1.60   113200.
N2  +  O   =  2 N +  O
0  3
                                                   1.11e+22   -1.60   113200.
N2  +  N2  =  2 N +  N2
0  3
                                                   3.70e+21   -1.60   113200.
N2  +  N   =  2 N +  N
0  3
                                                   1.11e+22   -1.60   113200.
NO  +  N   =  N  +  O  +  N
0  3
                                                   4.60e+17   -0.50    75500
NO  +  O   =  N  +  O  +  O
0  3
                                                   4.60e+17   -0.50    75500
NO  +  N2  =  N  +  O  +  N2
0  3
                                                   2.30e+17   -0.50    75500
NO  +  O2  =  N  +  O  +  O2
0  3
                                                   2.30e+17   -0.50    75500
NO  +  NO  =  N  +  O  +  NO
0  3
                                                   2.30e+17   -0.50    75500
```

```
NO  +    O  =  N   +  O2
0  3
                                                   2.16e+08    1.29    19220
O  +   N2  =  N   +  NO
0  3
                                                   3.18e+13    0.10    37700
N  +   O   =  NO+ +  E-
0  3
                                                   1.53e+10    0.37    32000
N  +   E-    =  2E-   +  N+
0  3
                                                   1.25e+35   -4.0    168600
O  +   E-    =  2 E-   +  O+
0  3
                                                   1.95e+34   -4.0    158500
```

A.2 Species Thermodynamics Coefficients

This section contains the default species thermodynamics database delivered with the HYLDA code. The database curve fits are based on the McBride curve fits.

```
Mcbride Thermodynamic Properties (Referenced to 0K)
11
E-
5.4858e-04    -1.4819    -1.
4
100.   1000.   6000.   20000.
0   7
 .25000000e+01    .00000000e+00   .00000000e+00   .00000000e+00   .00000000e+00
-.74537500e+03  -.11720829e+02
 .25000000e+01    .00000000e+00   .00000000e+00   .00000000e+00   .00000000e+00
-.74537500e+03  -.11720829e+02
 .25000000e+01    .00000000e+00   .00000000e+00   .00000000e+00   .00000000e+00
-.74537500e+03  -.11720829e+02
N+
14.00615    448.22    1.
4
100.   1000.   6000.   20000.
0   7
 .30042991e+01  -.29685010e-02   .67382302e-05  -.66959797e-08   .24287023e-11
 .22555768e+06   .27419129e+01
 .25159210e+01  -.14320456e-05  -.14038942e-07   .69253644e-11  -.62448531e-15
 .22562342e+06   .48976052e+01
 .19375382e+01   .22409829e-03  -.21648231e-07   .95031711e-12  -.15113875e-16
 .22657026e+06   .89458068e+01
N
14.0067    111.491    0.
4
100.     1000.   6000.   20000.
0   7
 .25000000e+01   .00000000e+00   .00000000e+00   .00000000e+00   .00000000e+00
 .56105322e+05   .41938876e+01
 .24159451e+01   .17488525e-03  -.11901918e-06   .30224818e-10  -.20359762e-14
 .56134458e+05   .46495767e+01
-.18329813e+01   .15387216e-02  -.15930668e-06   .72399936e-11  -.12163945e-15
 .64449569e+05   .35189994e+02
NO+
30.00555    234.693    1.
```

```
4
100.  1000.  6000.  20000.
0   7
 .36942311e+01  -.13500091e-02   .26890271e-05  -.10385219e-08  -.66335389e-13
 .11810386e+06   .30859671e+01
 .29453034e+01   .14042178e-02  -.49610822e-06   .79706799e-10  -.47279482e-14
 .11824551e+06   .67095724e+01
 .12994114e+02  -.30072159e-02   .33407414e-06  -.11484287e-10   .79537762e-16
 .99654632e+05  -.64176748e+02
NO
30.008   19.5632   0.
4
100.  1000.  6000.  20000.
0   7
 .42185268e+01  -.46390319e-02   .11041154e-04  -.93362475e-08   .28036107e-11
 .98447413e+04   .22808736e+01
 .32606447e+01   .11911186e-02  -.42917563e-06   .69458505e-10  -.40336582e-14
 .99210939e+04   .63693794e+01
 .78251966e+01  -.15947601e-02   .26157825e-06  -.15400836e-10   .29751352e-15
 .37039087e+04  -.24351732e+02
N2+
28.01285   358.652   1.
4
100.  1000.  6000.  20000.
0   7
 .37768643e+01  -.20751822e-02   .47847204e-05  -.31874735e-08   .68315816e-12
 .18048322e+06   .26877702e+01
 .36006071e+01   .21352419e-03   .21810644e-06  -.54061667e-10   .39111526e-14
 .18038982e+06   .30252535e+01
 .22053190e+01   .78003983e-03   .154934[illegible]e-08  -.46570141e-11   .14637220e-15
 .18400426e+06   .13325171e+02
N2
28.0134   -2.0805   0.
4
100.  1000.  6000.  20000.
0   7
 .35310053e+01  -.12366099e-03  -.50299944e-06   .24353061e-08  -.14088123e-11
-.10469763e+04   .29674747e+01
 .29525763e+01   .13969006e-02  -.49263169e-06   .78601037e-10  -.46075532e-14
-.92394864e+03   .58718925e+01
 .12298245e+02  -.29524883e-02   .36411038e-06  -.15492819e-10   .20126105e-15
-.17386647e+05  -.59539209e+02
O+
15.99885   372.335   1.
4
100.  1000.  6000.  20000.
0   7
 .25000000e+01   .00000000e+00   .00000000e+00   .00000000e+00   .00000000e+00
 .18793692e+06   .43933593e+01
 .24877348e+01   .21762768e-04  -.10893562e-07   .12586348e-11   .13734329e-15
 .18794159e+06   .44613143e+01
 .30796256e+01  -.43965601e-03   .92239825e-07  -.57035608e-11   .11238506e-15
 .18756475e+06   .73362750e+00
O
15.9994   57.870   0.
4
100.  1000.  6000.  20000.
0   7
 .31682720e+01  -.32793260e-02   .66430541e-05  -.61280412e-08   .21126471e-11
 .29122010e+05   .20518892e+01
 .25436369e+01  -.27315581e-04  -.41905464e-08   .49547498e-11  -.47954106e-15
 .29225764e+05   .49222771e+01
 .15374154e+01   .38907294e-03  -.45993406e-07   .24077395e-11  -.45167882e-16
 .30927600e+05  .11972685e+02
O2+
31.99825   277.695   1.
4
100.  1000.  6000.  20000.
0.  7
 .46101224e+01 -.63591540e-02  .14241607e-04 -.12098710e-07  .37091284e-11
 .13974364e+06 -.20112185e+00
 .33167625e+01  .11152214e-02 -.38349274e-06  .57278609e-10 -.27765010e-14
 .13987822e+06  .54472465e+01
 .35444851e+01 -.63072734e-03  .24818766e-06 -.20004272e-10  .46468638e-15
 .14445694e+06  .70858389e+01
O2
31.9988   -2.1143   0.
4
100.  1000.  6000.  20000.
0   7
 .37825246e+01 -.29962626e-02  .98439984e-05 -.96778469e-08  .32429006e-11
-.10639621e+04  .36569569e+01
 .36645375e+01  .64809452e-03 -.13493979e-06  .18738921e-10 -.11169387e-14
-.12171837e+04  .33959041e+01
 .17191663e+01  .13171091e-02 -.16019927e-06  .73852088e-11 -.11923265e-15
 .23584342e+04  .17254483e+02
```

A.3 Transport Properties Coefficients

This section contains the default transport properties database delivered with the HYLDA code. The databas. is based on the Nicolet transport property model for viscosity, thermal conductivity, and species diffusion coefficients.

```
NICOLET TRANSPORT PROPERTY DATA (MU, LAMBDA, DIFF. COEFF.)
1
9  0
E-
9
0  4
E-
    1.0000     0.961      1.0000     0.461
N+
    1.0000     0.961      1.0000     0.461
N
     0.0000     1.609      0.0000     1.609
NO+
   1.0000     0.961      1.0000     0.461
NO
    0.5322     1.308      0.5322     1.308
N2
    0.2870     1.841      0.2870     1.841
O+
    1.0000     0.961      1.0000     0.461
O
     0.6759    -0.5547     0.6759    -0.5447
O2
```

```
     0.4748    0.9083    0.4748    0.9083
N+
9
0  4
E-
     1.0000    0.961     1.0000    0.461
N+
     1.0000    0.961     1.0000    0.461
N
    -0.1010    3.970    -0.3568    3.726
NO+
     1.0000    0.961     1.0000    0.461
NO
    -0.2048    3.219    -0.1679    3.367
N2
    -0.3128    3.262    -0.2739    3.434
O+
     1.0000    0.961     1.0000    0.461
O
    -0.3979    4.094    -0.3999    4.007
O2
    -0.3979    4.094    -0.3999    4.007
N
9
0   4
E-
     0.0000    1.609     0.0000    1.609
N+
    -0.1010    3.970    -0.3568    3.726
N
    -0.3098    2.996    -0.2817    3.091
NO+
    -0.2048    3.219    -0.1679    3.367
NO
    -0.2048    3.219    -0.1679    3.367
N2
    -0.3128    3.262    -0.2739    3.434
O+
    -0.3979    4.094    -0.3999    4.007
O
    -0.3424    3.091    -0.3327    3.243
O2
    -0.2872    3.329    -0.2722    3.512
NO+
9
0   4
E-
     1.0000    0.961     1.0000    0.461
N+
     1.0000    0.961     1.0000    0.461
N
     -0.2048    3.219    -0.1679    3.367
NO+
   1.0000     0.961      1.0000     0.461
NO
   -0.1269     4.291     -0.3979     3.750
N2
   -0.1859     3.367     -0.1383     3.497
O+
     1.0000     0.961      1.0000     0.461
O
     -0.2529     3.243     -0.2074     3.384
O2
   -0.1549     3.367     -0.1120     3.497
NO
9
0   4
E-
    0.5322     1.308      0.5322     1.308
N+
   -0.2048     3.219     -0.1679     3.367
N
     -0.2048     3.219     -0.1679     3.367
NO+
 -0.1269     4.291     -0.3979     3.750
NO
   -0.1461     3.307     -0.1359     3.512
N2
   -0.1859     3.367     -0.1383     3.497
O+
   -0.2529     3.243     -0.2074     3.384
O
     -0.2529     3.243     -0.2074     3.384
O2
   -0.1549     3.367     -0.1120     3.497
N2
9
0   4
E-
    0.2870     1.841      0.2870     1.841
N+
   -0.3128     3.262     -0.2739     3.434
N
     -0.3128     3.262     -0.2739     3.434
NO+
 -0.1859     3.367     -0.1383     3.497
NO
   -0.1859     3.367     -0.1383     3.497
N2
   -0.2739     3.434     -0.2613     3.597
O+
   -0.2872     3.329     -0.2722     3.512
O
     -0.2872     3.329     -0.2722     3.512
O2
   -0.1549     3.367     -0.1120     3.497
```

```
O+
9
0   4
E-
    1.0000    0.961     1.0000    0.461
N+
    1.0000    0.961     1.0000    0.461
N
    -0.3979    4.094     -0.3999    4.007
NO+
   1.0000    0.961     1.0000    0.461
NO
   -0.2529    3.243    -0.2074    3.384
N2
   -0.2872    3.329    -0.2722    3.512
O+
    1.0000    0.961     1.0000    0.461
O
    -0.0860    4.159     -0.3657    3.645
O2
   -0.2389    3.153    -0.2219    3.314
O
9
0   4
E-
    0.6759   -0.5547    0.6759   -0.5447
N+
   -0.3979    4.094    -0.3999    4.007
N
    -0.3424    3.091     -0.3327    3.243
NO+
  -0.2529    3.243    -0.2074    3.384
NO
   -0.2529    3.243    -0.2074    3.384
N2
   -0.2872    3.329    -0.2722    3.512
O+
   -0.0860    4.159    -0.3657    3.645
O
    -0.2601    2.955     -0.2632    3.140
O2
   -0.2389    3.153    -0.2219    3.314
O2
9
0   4
E-
    0.4748    0.9083    0.4748    0.9083
N+
   -0.3979    4.094    -0.3999    4.007
N
   -0.2872    3.329    -0.2722    3.512
NO+
  -0.1549    3.367    -0.1120    3.497
```

NO	-0.1549	3.367	-0.1120	3.497
N2	-0.1549	3.367	-0.1120	3.497
O+	-0.2389	3.153	-0.2219	3.314
O	-0.2389	3.153	-0.2219	3.314
O2	-0.1503	3.296	-0.1166	3.434

☆U.S. Government Printing Office:1992—648-127/62295